Leveraging LLM For Synchronizing Information Across Multilingual Tables

Siddharth Khincha[1], **Tushar Kataria**[2] **Ankita Anand**[1], **Dan Roth**[3], **Vivek Gupta**[4*]
[1]IIT Guwahati, [2]University of Utah
[3]University of Pennsylvania, [4]Arizona State University
{s.khincha,ankita.anand}@iitg.ac.in, tkataria@cs.utah.edu, danroth@seas.upenn.edu, vgupt140@asu.edu

Abstract

The vast amount of online information today poses challenges for non-English speakers, as much of it is concentrated in high-resource languages such as English and French. Wikipedia reflects this imbalance, with content in low-resource languages frequently outdated or incomplete. Recent research has sought to improve cross-language synchronization of Wikipedia tables using rule-based methods. These approaches can be effective, but they struggle with complexity and generalization. This paper explores large language models (LLMs) for multilingual information synchronization, using zero-shot prompting as a scalable solution. We introduce the *Information Updation* dataset, simulating the real-world process of updating outdated Wikipedia tables, and evaluate LLM performance. Our findings reveal that single-prompt approaches often produce suboptimal results, prompting us to introduce a task decomposition strategy that enhances coherence and accuracy. Our proposed method outperforms existing baselines, particularly in Information Updation (1.79%) and Information Addition (20.58%), highlighting the model's strength in dynamically updating and enriching data across architectures.

1 Introduction

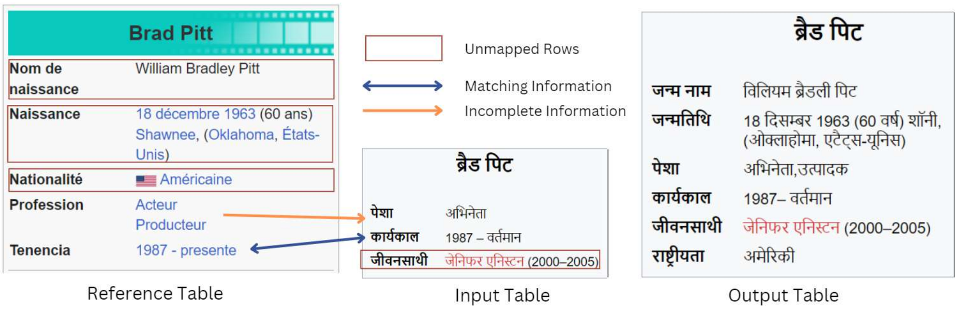
In today's digital era, nearly every subject/domain is discoverable online. With global access to high-speed internet expanding, the volume of information grows exponentially[1][2]. From movies and celebrities to elections and corporate news, a vast array of topics is just a click away for those with access. However, since developed countries—particularly English-speaking ones—were early adopters of the internet, much online content is tailored to English-speaking audiences[3]. This is evident on platforms such as Wikipedia and YouTube, where English dominates[4]. Although the number of non-English users is growing, under-represented languages such as Afrikaans, Cebuano, and Hindi still face a significant information gap (Bao et al., 2012).

As shown by Khincha et al. (2023) in their case study on Wikipedia's entity-centric tables, information in Wikipedia infoboxes (Zhang and Balog, 2020) is heavily skewed toward high-resource languages such as English, Spanish, and French. They found that tables in low-resource languages often lack key information and are frequently outdated or inaccurate (Jang et al., 2016; Nguyen et al., 2018). This disparity is especially concerning in the digital age, where misinformation on widely accessible platforms can have far-reaching consequences. To address this, Khincha et al. (2023) developed the InfoSync dataset to analyze information synchronization issues across 14 languages. They proposed aligning tables by matching similar keys and using rule-based methods to transfer and update information in tables across languages. Figure 1 illustrates an example of information synchronization across multilingual tables (Spanish and Hindi). However, InfoSync's approach has a key limitation: rule-based methods become increasingly complex as new corner cases emerge, making generalization challenging. A more effective alternative is leveraging current large language models (LLMs) for zero-shot prompting, providing an easily scalable solution for these tasks.

With new and advanced LLMs (such as GPT, Mitral, LLAMA (Brown et al., 2020; Touvron et al., 2023; Achiam et al., 2023; Jiang et al., 2023)) being released every year, the zero-shot prompting capabilities of these models are improving with each new training iteration. These LLMs are consistently approaching, and in some cases surpassing, human performance across various NLP ap-

[*] Corresponding Author
[1] Statista Worldwide Internet Usage
[2] Cisco Annual Internet Report
[3] Wikipedia: List of Wikipedians by Number of Edits
[4] Wikipedia: Active Users by Language

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 6474–6492
April 29 - May 4, 2025 ©2025 Association for Computational Linguistics



Brad Pitt	
Nom de naissance	William Bradley Pitt
Naissance	18 décembre 1963 (60 ans) Shawnee, (Oklahoma, États-Unis)
Nationalité	Américaine
Profession	Acteur Producteur
Tenencia	1987 - presente

ब्रैड पिट	
पेशा	अभिनेता
कार्यकाल	1987 – वर्तमान
जीवनसाथी	जेनिफर एनिस्टन (2000–2005)

ब्रैड पिट	
जन्म नाम	विलियम ब्रैडली पिट
जन्मतिथि	18 दिसम्बर 1963 (60 वर्ष) शॉनी, (ओक्लाहोमा, एटैट्स-यूनिस)
पेशा	अभिनेता,उत्पादक
कार्यकाल	1987– वर्तमान
जीवनसाथी	जेनिफर एनिस्टन (2000–2005)
राष्ट्रीयता	अमेरिकी

Figure 1: Example of information synchronization across multilingual tables. A reference table in a high-resource language is used to update outdated input tables in a low-resource language, resulting in an updated output table in the low-resource language.

plications (Achiam et al., 2023). LLMs excel at text-generation (Achiam et al., 2023), text modification based on provided prompts (Raffel et al., 2020), and text refinement by correcting errors (Davis et al., 2024; Liu et al., 2024; Li et al., 2024). Given the substantial advancements in LLM capabilities, this paper poses the following questions: *Can LLMs be leveraged for information synchronization in multilingual Wikipedia entity-centric tables? If so, how can LLMs be utilized, and how effective are they compared to the rule-based methods proposed by Khincha et al. (2023)?*

To investigate these questions, we first construct an *Information Updation* dataset. This dataset simulates the information updating task by using outdated versions of tables to represent old information, and comparing them with the latest versions that have been manually corrected to address any missing or outdated content. Additionally, we assess the performance of state-of-the-art large language models (LLMs), specifically GPT-4 (Achiam et al., 2023), for the task of information synchronization in entity-centric tables across multiple languages. To evaluate performance on the *Information Updation* task, we propose novel automated metrics that offer valuable insights into model performance and identify potential areas for improvement in future iterations.

Our initial experiments reveal that relying on a single prompt for multilingual information synchronization with LLMs yields suboptimal performance, frequently producing incoherent edits. To enhance these results, we propose a task decomposition approach. Our proposed method uses multiple prompts to address smaller, more manageable subtasks, which are then connected in a sequential pipeline to generate the final output. Task decomposition has shown promise in improving performance across a range of complex NLP applications (Khot et al., 2022; Ma et al., 2024; Wang et al., 2024), and we find it similarly beneficial in our work. Our work makes the following contributions:

- We create an *Information Updation* dataset by sampling older versions of the same Wikipedia pages. This dataset simulates the real world process of updating Wikipedia infoboxes with human input, reflecting the task of correcting and adding new information over time.
- We employ large language models (LLMs) for zero-shot automated multilingual information synchronization in entity-centric Wikipedia tables. By utilizing prompt-based task decomposition, we significantly enhance the accuracy and coherence of the results.
- Develop novel evaluation metrics for the Information Updation task, alongside conducting a thorough error analysis to identify the limitations of current state-of-the-art LLM models.

Code and dataset are available at https://zero-shot-llm-infosync.github.io/zero-shot-llm-infosync/.

2 Proposed Methodology

Information synchronization for Wikipedia infoboxes involves updating outdated rows in the table by conditionally modifying attributes, values, or both, using data from reference infoboxes(which have updated information). Consider a source table (T_S) in language (L_i) that contains missing or outdated information. We also have a reference table (T_R) in language (L_j), which provides the missing and updated information not found in T_S. Additionally, we assume access to gold-standard updated table (T_G) in language (L_i), which can be regarded as having been manually curated.

The *Information Synchronization* task is to update the table T_S using only the information available in T_S and T_R, with the goal of matching the updated table to T_G. Previous work by Khincha et al. (2023) approaches this problem with a two-step methodology: (i) *Information Alignment*, which involves identifying similar rows across different tables using cosine similarity, and (ii) *Information Update*, which utilizes rule-based methods to update T_S. In contrast, we propose to solve this task using large language models (LLMs) to provide a more automated and sophisticated solution, bypassing the need for elementary similarity measures and rule-based methods. We propose solving the information synchronization problem using various prompts, as outlined below:

Simple Prompt. With large-scale pre-training of language modeling, new language models such as GPT4, LLaMA, and Gemini Pro support prompt-based (instruction set) evaluations, which do not require any finetuning. As a baseline prompt, we explain the task of information synchronization in the prompt giving details of types of missing information that might be presented between the two tables such as outdated information, missing information, or inconsistent information. The model is tasked to create an output table that has updated information from both source and reference tables. Here, we ask the model to give more importance to reference table information while creating an updated output table.

Elementary Task decomposition within a Single Prompt: To test whether a single prompt can give reasonable outputs even when directed to do task decomposition as an intermediate step, we propose *Align-Update Decomposition* prompt. In this prompt, the model is instructed to first implicitly align all corresponding information between the two tables. Once these alignments are automatically generated, the model should carefully review each alignment to identify and remove any outdated information wherever necessary. Additionally, the model is explicitly instructed to add missing rows that could not be mapped during the alignment process. This prompt is inspired by the task decomposition of Khincha et al. (2023), which does the same with rule-based approach.

Hierarchical Task Decomposition Prompt. Instead of creating a single instruction set for the task of Information synchronization. We do a hierarchical decomposition of the task and create prompts for each step. These prompts are applied sequentially, with the output of the last prompt as input to the next prompt in the hierarchy. Different hierarchical steps for this prompt are:

- **Translation**: All tables (T_S, T_R, and T_G) from different languages are converted to English. English is selected as the base language because most state-of-the-art LLMs are largely trained on curated English data, resulting in higher accuracy for complex reasoning and analysis tasks performed in English compared to other languages.
- **Knowledge graphs conversion**: The translated source and reference tables are then converted into knowledge graphs. Our experiments indicate that the subsequent hierarchical steps are more effective when using knowledge graphs rather than infoboxes/tables. LLMs perform better reasoning over knowledge graphs.
- **Merging or alignment**: The source(KG_S) and reference(KG_R) knowledge graphs are merged to create a unified knowledge graph that consolidates all the information from both sources. Merging of knowledge graph is equal to alignment step described in the section above. This merging process helps eliminate redundant information from unresolved conflicts and enhances the inclusion of missing details. During the merge step, the model first gathers all necessary information, and in the subsequent update step, it makes the relevant adjustments.
- **Update:** The merged knowledge graphs is used to update information in the source knowledge graphs. Due to these being node operations, these are fast and have better interpretablity.

After the update step, the revised knowledge graphs are converted back into tables. These tables are then translated back into the original languages of the source tables. We compare these three prompt designs for the task of information synchronization with relevant ablations for heirarchical task decomposition. Prompt examples are shown in Appendix B.

3 InfoUpdate Benchmark

Khincha et al. (2023) concentrated on developing a dataset for information alignment tasks, i.e., aligning similar keys across tables coming from different languages. They employed a rule-based method for updates and conducted human evaluations based on these updates to recommend edits on Wikipedia pages. However, they did not create a dataset or propose an automated method for evaluating approaches to information updation. To address this, we introduce a new human-annotated dataset (InfoUpdate) for the *Information Updation* task focused on Wikipedia infoboxes. This dataset comprises approximately 950 annotated instances across 9 categories: Album, Athlete, City, College, Company, Country, Musician, Person, and Stadium, spanning 14 languages including Spanish, French, German, Arabic, Hindi, Korean, Russian, Afrikaans, Cebuano, Swedish, Dutch, Turkish, and Chinese. Additional dataset statistics are shown in Appendix Table 4.

InfoUpdate **Construction.** We construct the dataset by extracting two versions of the same Wikipedia table entity from different time periods. For a table in Category C titled T and language(L_i), we extract the *Old* version from 2018 (source table) and the *New* version from 2023 (current at the time of extraction). The new version of the table is extracted from two or more different languages, where the table in the same language serves as the target, and the table in a different language acts as a reference (or additional information), i.e., updated table. This setup is designed to simulate a real-world implementation of the Information Synchronization task for entity centric semistructured tables. For every entity in a single instance of the task, the dataset contains 3 tables source table T_S, reference table T_R and a gold table T_G.

- The **Source Table** T_S is the *(*Old) outdated version of the entity in language L_i.
- The **Reference Table** T_R is the *New* updated version of the entity in language $L_j (i \neq j)$
- The **Gold Table** T_G is the human annotated version, which is manually created by synchronising the *New* updated versions of the entity in the languages L_i and L_j.

The information updation task is to update rows in source table (T_S) using reference table (T_R) as context information, so that the resulting generated table, referred to as the output table (T_O), has the same information as Gold Table (T_G).

InfoUpdate **Verification.** Human annotators are tasked to ensure two aspects:(a) The gold table contains the complete information present in both the input and reference table combined without any redundancy. (b) The gold table is consistent, resolving all conflicts or missing data without adding new information not found in the Source or Reference tables.

4 InfoUpdate Evaluation Metric

For evaluation, we first experimented with an approach similar to Chiang and Lee (2023), where we tasked the model with comparing the output table against a reference table and providing a score. However, this approach proved to be highly unstable, resulting in vastly different responses even in low-temperature settings. Moreover, the model often overlooked multiple rows during evaluation, making it difficult to establish a consistent scoring system due to the subjective nature of the process. Therefore, we propose our novel evaluation metric consisting of two main steps: (a) *Information Alignment Evaluation* and (b) *Information Update Evaluation*, aimed at addressing the aforementioned issues.

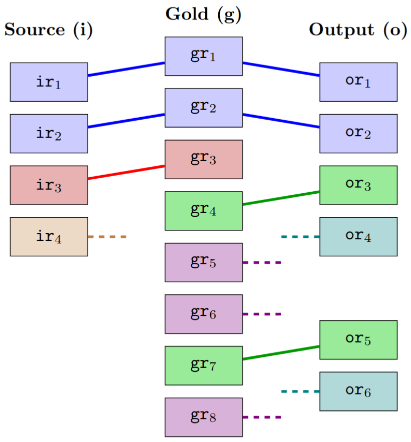


Figure 2: **Alignment Groups For Information Alignment**. All rows highlighted in blue and connected by blue lines in the Source, Gold, and Output tables are tri-aligned, meaning they contain the same information across all three tables. Rows highlighted in red or green are bi-aligned, indicating that the information is consistent either between the Input and Gold tables or the Gold and Output tables. The remaining rows are unaligned, containing differing information.

Information Alignment Evaluation. In this step, we first create a mapping of similar information or alignments between {Source Table(T_S), Gold Table(T_G)} and {Output Table (Generated Table)(T_O), Gold Table(T_G)} tables. Reference table (T_R) is not used during the evaluation process, as it in different language and only used as referenced for updating source (T_S) tables.

Alignment between (T_S) and (T_G) gives us a metric of *information already present in the source when compared to gold*, whereas alignment between (T_O), and (T_G) informs us of *extra alignments added due to the information generated by the synchronization model, i.e., table updation*. The alignments are compared and separated into three groups: Tri-Aligned, Bi-Aligned, and Un-Aligned. Figure 2 shows pictorial the meaning and different between the three alignment group we defined and Table 5 shows the results for each types of alignments. Formal definitions in logic statements can be found in Appendix Table 6.

- **Tri-Aligned**: These refer to the table keys that are common across all three tables: the SOURCE, OUTPUT, and GOLD. (ir_1, gr_1, or_1) and (ir_2, gr_2, or_2) are trialigned rows in example Figure 2. These represent information that is either kept intact by the model from source to output, or can also have cases where the information was incomplete in source table but was completed by model operations.
- **Bi-aligned**: When the table keys are common across pairs of tables GOLD-OUTPUT $(gr_4, or_3), (gr_7, or_5)$ or GOLD-SOURCE (ir_3, gr_3), but not across all three tables, these define Bi-aligned rows. The number of Gold elements aligned with the Output but not the Input indicates the amount of information added, which was not present before $(gr_4, or_3), (gr_7, or_5)$, while the number of Gold elements aligned with the Input but not the Output represents the amount of relevant information deleted (ir_3, gr_3).
- **Un-aligned**: These are keys remaining in SOURCE, OUTPUT and GOLD tables after tri- and bi-aligned keys are removed from tables. We have three types of unaligned rows as follows:

(a.) *Unaligned* SOURCE TABLE *keys* (ir_4), refers to redundant input information deleted in the output table.

(b.) *Unaligned* OUTPUT TABLE *keys* (or_4, or_6), refers to hallucinated/noisy/irrelavant information added to output table not present in either (Source or Gold).

(c.) *Unaligned* GOLD TABLE *keys* (gr_5, gr_6, gr_8), refer to information gaps that the model could not add to the Output table, either due to model inaccuracies or because the information is missing in the Source and Reference tables.

Information Updation Evaluation. We evaluate each alignment pair from both SOURCE-GOLD alignments and OUTPUT-GOLD alignments for semantic equivalence using Large Language Models (Eval$_{\text{LLM}}$). Here, we check if the align information is fully-matching, partially matching or contradictory to each other. For each aligned key-value pair, the LLM is instructed to examine the information, translate it into English, and decompose it into fine-grained atomic details. These atomic details are then categorized into four distinct groups: (a) **Similar and Consistent (SCT)**—information appearing in both tables with consistent values; (b) **Similar and Contradictory (SCD)**—information present in both tables but exhibiting contradictory or conflicting values; (c) **Table 1 Unique (T1U)**—information unique to Table 1, not found in Table 2; and (d) **Table 2 Unique (T2U)**—information unique to Table 2, not present in Table 1. These four categories are used to calculate the precision and recall for evaluation as follows:

$$\text{Precision} = \frac{|\text{SCT}|}{|\text{SCT}| + |\text{SCD}| + |\text{T1U}|}$$

$$\text{Recall} = \frac{|\text{SCT}|}{|\text{SCT}| + |\text{SCD}| + |\text{T2U}|}$$

$$\text{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Here, | X | denotes the cardinality of the set X, indicating the number of elements within set X. The precision and recall scores are normalized by dividing them by the length of the gold table (T_G), ensuring a fair comparison. We use these measures to evaluate **SCT**, **SCD**, **T1U** and **T2U** for different models. We used this metric for all information alignment. The detail metric for each information alignment types (tri, bi, and un) is shown in Appendix Table 7, grounded with the running examples shown in the Figure 2.

5 Experiments and Results

Alignment Models. For *information alignment*, we employ an ensemble of multiround voting methods that combine InfoSync—a deterministic alignment algorithm—with LLM-based alignment utilizing few-shot chain-of-thought prompting with detailed instructions. We conducted three runs each with GPT-3.5 and Gemini 1.5 Flash Pro, followed by majority voting to establish the LLM alignment. The majority-voted alignments from both GPT-3.5 and Gemini 1.5 Flash Pro are then integrated into InfoSync, using majority voting again to achieve the final alignment. We refer to this approach as the multi-voting scheme.

Updation Models. For *information updation*, we utilize three large language models (LLMs) for our experiments: GPT-3.5, Gemini 1.5 Flash Pro (Reid et al., 2024), and LLAMA 3.0 (70B) (AI@Meta, 2024). The first two are closed-source models accessible only via API, whereas the latter is an open-source model. These models were selected as they represent state-of-the-art performance or are at least close to it and fall within the computational budget of our project. We believe that our results will also be applicable to other closed and open-source LLMs.

Evaluation Strategy. Evaluating the efficacy of large language models (LLMs) involves two key steps: (a) information alignment, which prepares the tables for comparison, and (b) information evaluation, which assesses the semantics. We evaluate both of these as following:

1. Alignment Evaluation. The alignments are compared with human alignments for both INPUT-OUTPUT (200 pairs) and OUTPUT-GOLD (200 pairs), for GPT 3.5 generated updated tables with our decomposition approach.

2. Updation Evaluation. Human evaluation is the ideal approach; however, it is extremely cumbersome and costly. Therefore, we use an LLM-based evaluation to assess the effectiveness of information updates. We utilized average outputs from three closed-source LLMs accessed via their APIs: Gemini 1.5 Flash Pro, GPT-4, and GPT-3.5. [5] In our evaluation process, we systematically measured the individual similarity of each aligned row for both INPUT-OUTPUT and OUTPUT-GOLD across all three models. After generating prediction scores, we averaged the similarity scores from each model to obtain a consolidated view of overall alignment quality. This ensemble approach proved significantly more effective for similarity matching than relying on a single LLM. By averaging the output of multiple models, we effectively utilized their diverse strengths, enhancing the robustness and accuracy of our semantic evaluations. This also mitigated the limitations associated with any individual model and provided a more consistent semantic matching.

Baselines Methods. We compare our proposed decomposition approach against multiple baselines, including both *deterministic rule-based* and *generation-based methods*. For the deterministic rule-based comparison, we utilize the rule-based update technique from InfoSync (Khincha et al., 2023). In the generation-based approach, we take a straightforward approach with direct prompting. We simply describe the task at hand, without breaking it into smaller steps or outlining modeling strategies. The model is then free to tackle the task on its own, using chain-of-thought reasoning to guide its process. Additionally, we adapt the InfoSync technique (Align-Update variants, two and joint prompts), implementing a two-step process involving initial alignment followed by updates with a large language model (LLM). This involves two strategies: one where we provide instructions in a single prompt (joint prompts) to perform both steps simultaneously, and another where we use two sequential prompts—one (two prompts) for alignment and the output of which feeds into a second prompt that handles the final updates.

Ablations. Additionally, we also compare our approach through various ablation studies, starting with a single prompt containing all instructions, referred to as the Direct Decompose Prompt. We then conduct step-wise ablations of our decomposition method, beginning with just English translation and direct updates, followed by back-translation, which we refer to as Translation (+BackTrans). Next, we incorporate the merge and alignment steps in place of direct updates, referred to as Merge and Alignment. Finally, we transform the output into a knowledge graph before merging and aligning, completing our decomposition methods, which we refer to as Knowledge Graphs.

Human Baseline. We also assess our model's performance against a human baseline (from independent research team members), although this evaluation is limited to only 100 randomly selected table updating pairs due to cost and time constraints.

5.1 Results and Analysis

Information Alignment Results. The results of Alignment are shown in Table 2, which show that our multi-voting achieves superior alignment, demonstrating very high precision and recall, with an overall F1 score of 93%. Additionally, multi-voting enhances robustness and reduces variations inherent in each individual model, leading to a more consistent and reliable alignment outcome.

Information Updation Results. Our main results,

[5] We avoided open-source models like LLaMA 3.0 due to their weaker performance in semantic similarity tasks, as prior research shows closed-source models excel in capturing subtle semantics.

	Trialign Rows (Tr)	Bialign Rows (Bi)		UnAlign Gold (UG)	Input BiAlign (Bi)
Methods	Updated ↑	Added (%) ↑	Added (#Rows) ↑	Missed (G) ↓	Delete (I) ↓
InfoSync (Khincha et al., 2023)	1.28	12.18	2.99	4.67	0.35
Direct Prompt	0.63	11.55	3.63	4.40	0.50
Align-Update (Two Prompts)	-0.77	12.59	3.98	2.74	**0.14**
Align-Update (Joint Prompt)	0.51	13.58	3.48	3.24	0.17
Our Proposed Decomposition Approach					
Direct Decompose Prompt	0.90	12.06	2.98	4.65	0.35
Translation(+BackTrans)	0.62	16.88	4.09	3.71	0.38
+ Merge and Alignment	1.33	17.80	**4.99**	2.92	0.48
+ Knowledge Graph	**1.79**	**20.58**	4.88	**2.69**	0.45
Human (100 examples)	1.75	21.44	5.6	2.09	**0.12**

Table 1: **Information updation results with average over multiple LLMs.** The performance is reported after using the average of similarity score for multiple LLMs for *information evaluation*, including GPT-3.5, LLaMA 3.0 (70B), and Gemini 1.5 Flash Pro. These results also include ablation studies on various components of our proposed task decomposition, including Translation, KG conversion, and merge-alignment.

Model	**Type**	**Precision**	**Recall**	**F1**
InfoSync	Input_Gold	96.62	88.64	91.26
	Output_Gold	89.37	82.07	84.42
	Overall Average	92.90	85.27	87.75
GPT3.5	Input_Gold	96.29	93.99	94.40
	Output_Gold	89.63	85.98	86.70
	Overall Average	92.88	89.88	90.46
GPT3.5 voting(3x)	Input_Gold	98.06	**94.10**	95.66
	Output_Gold	**94.57**	87.41	89.81
	Overall Average	96.27	90.67	92.66
Gemini voting(3x)	Input_Gold	96.88	92.05	93.63
	Output_Gold	92.52	83.95	86.46
	Overall Average	94.65	87.90	89.96
Multi voting(3x)	Input_Gold	**99.15**	93.88	**95.80**
	Output_Gold	94.18	**88.39**	**90.69**
	Overall Average	**96.60**	**91.07**	**93.18**

Table 2: Evaluation models alignment performance.

i.e. on updating information, are shown in Table 1, demonstrate that our proposed decomposition technique significantly outperforms several baselines, including Khincha et al. (2023), particularly in Information Updation (1.79%) and Information Addition (20.58%), highlighting the methods strength in dynamically updating and enriching data.

Our approach excels in correcting erroneous and adding missing information, consistently outperforming other methods. For instance, Align-Update (Two Prompts) shows a negative result (-0.77) in updates, while our approach performs reliably across key metrics. It captures missing data effectively, with the least amount of missed rows (2.69), closest to human performance (2.09 rows). Although there is a slight increase in the deletion rate (0.45) due to more prompts, this is outweighed by the improvements in adding and updating information, demonstrating our method's strength.

The integration of the +*Knowledge Graph* plays a crucial role in boosting performance, particularly in Information Addition, where we achieve the highest score of 20.58%. The combination of +*merging* and alignment techniques further enhances performance, reflected in the added rows metric (4.99). The model falls slightly behind human performance in deletion rates, but it outperforms in information addition and updating, significantly narrowing gap between human and model.

We observe consistent performance gains across models, with detailed results in Table 11 in Appendix A.3 for GPT 3.5, Gemini 1.5 flash Pro, and LLAMA 3.0 (70B). Our approach consistently outperforms existing methods across architectures. Gemini 1.5 Flash Pro achieves the highest scores, excelling in complex tasks. LLAMA 3.0 performs well but slightly behind Gemini 1.5, whereas GPT-3.5 improves on earlier baselines but lags behind the others. These differences likely reflect variations in architecture and training data, with Gemini's advanced features providing an edge. Our method's consistent success across models highlights its broad effectiveness and generalizability.

Where do LLM Fail? We performed a stepwise error analysis of our approach compared to the Gold Tables, classifying the errors into the following categories: **(a) Missing Information**: A complete row is missing from the table compared to the gold table. **(b) Outdated Information (Full)**: The entire row contains outdated information when compared to the gold table. **(c) Outdated Information (Partial)**: Some parts of the row are outdated, while others are up to date. **(d) Redundant Information**: The model retains redundant data from the input alongside updated information from the reference. This categorization helps identify the specific areas where the model's performance di-

Error Types.	In Refer.	+Tr. (En)	+ KG Cons.	+ Merge	+ Table Conv.	+ Tr. (BT-Orig)
Missing	145	145	**151 (+6)**	**198(+47)**	**202 (+4)**	202
Outdated (Full)	35	35	35	**51(+16)**	**59(+8)**	59
Outdated (Partial)	59	59	59	**68 (+9)**	**73 (+4)**	73
Redundant	0	0	**66 (+66)**	66	66	66
Total	239	239	**311 (+72)**	**383 (+72)**	**400 (+17)**	400

Table 3: **Error analysis:** Step-wise error analysis of the decomposition pipeline, showing error compounding at each stage: "In Reference" refers to total errors in the input reference tables (*lower bound*), "+Tr. (En)" captures total errors after translation to English, "+KG Cons." indicates total errors after knowledge graph construction, "+Merge" shows total errors after merging, "+Table Conv." tracks total errors after converting graphs to tables, and "+Tr. (BT-Orig)" refers to total errors after back translation. Numbers in parentheses reflect incremental error increases at each stage.

verges from the gold standard.

Table 3 breaks down the errors introduced at various stages of our decomposition pipeline, categorizing them into four types: Missing Information, Outdated Information (Full), Outdated Information (Partial), and Redundant Information. Each column tracks errors at different stages: "In Reference" shows baseline errors in the reference tables which are used to update outdated tables. These errors serve as a lower bound because even a perfect method cannot resolve information that is missing from the reference table used for updates. The reference tables initially contain 239 errors, including 145 missing rows, 35 fully outdated rows, and 59 partially outdated rows, with no redundancy.

The subsequent columns detail total errors after each step, i.e. translation to English (+Tr. En), knowledge graph construction (+KG Cons.), merging (+Merge), table conversion, i.e., restructuring (+Table Conv.), and back translation to the original language (+Tr. BT-Orig). The numbers in parentheses show the errors added in that step as compared to the prior step, i.e. new errors introduced in that step. Translation to English adds no new errors. However, errors rise to 400 after the knowledge graph construction and merging stages. Knowledge graph construction introduces 6 missing rows and 66 redundant rows, highlighting issues with removing outdated data. The merging stage adds 47 missing rows, 16 fully outdated rows, and 9 partially outdated rows, indicating challenges with data integration. Improving these stages, particularly knowledge graph construction and merging, could greatly reduce errors and enhance accuracy.

6 Further Discussion

Why Zero-Shot over Few-Shot ? Our proposed method employs hierarchical decomposition to break the problem into multiple, simplified tasks, effectively eliminating the reliance on few-shot learning. Moreover, using a few-shot approach may introduce bias, as the model could perform disproportionately well on categories selected as exemplars. To ensure that information synchronization remains unbiased and generalizable across various entity categories, we opted for a zero-shot setting. Lastly, our preliminary tests revealed that the increased computational and financial costs associated with few-shot learning using GPT APIs resulted in only marginal improvements, making it a less practical choice for our objectives.

Why Translation to English? Translation models for low-resource languages like Afrikaans and Cebuano often struggle with accuracy. Our research shows that direct translations between these languages tend to be unreliable, but translating everything into English boosts performance significantly. To address cultural nuances, we employ a two-way translation strategy. For reverse translations, we include examples from the original English-to-X translations, formatted as X-to-English mappings, as few-shot prompts. This approach helps capture cultural context and ensures more accurate alignment without losing meaning, especially for complex terms and idioms. This trend aligns with the fact that large language models (LLMs) are predominantly trained on English data. Although multilingual models continue to improve, LLMs trained primarily with English datasets still excel at tasks like knowledge graph construction, data merging, and table generation. Translating content into English allows us to leverage these LLMs' capabilities, improving semantic accuracy, consistency, and cross-lingual data processing. Additional discussion on explanation for using Knowledge graphs in the above proposed methods is explained in Appendix section A.4, showing an example.

7 Related Works

MultiLingual Information Alignment and Update. Past efforts in multilingual table attribute alignment have employed both supervised and unsupervised techniques. Supervised methods used simple classifiers based on features such as cross-language links and cosine text similarity derived from tables (Adar et al., 2009; Zhang et al., 2017; Ta and Anutariya, 2015). On the other hand, unsupervised approaches relied on corpus statistics and template or schema matching for alignment (Bouma et al., 2009; Nguyen et al., 2011). Previous research on information updates (Iv et al., 2022; Spangher et al., 2022; Panthaplackel et al., 2022; Zhang et al., 2020a,b) has primarily focused on Wikipedia and news articles, rather than semistructured data like tables. Spangher et al. (2022), specifically, examines the challenge of updating multilingual news articles across different languages. The most closely related work is (Khincha et al., 2023), which proposed a rule-based approach. However, this method struggles with corner cases and does not leverage current state-of-the-art large language models (LLMs) for multilingual information synchronization. Our work addresses these gaps by introducing an LLM-based prompting approach that is adaptable across different languages, providing a more scalable solution.

Temporal Understanding. Temporal evolving information has been explored through various datasets in the context of question answering. TORQUE (Ning et al., 2020) and TIME-SENSITIVEQA (Chen et al., 2021) focus on time-sensitive questions from Wikipedia, while SYGMA (Neelam et al., 2022), CRONQUESTIONS (Saxena et al., 2021), and TEMPQUESTIONS (Jia et al., 2018) deal with temporal queries in knowledge graphs. SUMIE (Hwang et al., 2024), addresses a similar task in a specific domain and shares some similarities with our work. SUMIE deals with textual summarization, analyzing unstructured text. In contrast, our approach focuses on semistructured data, specifically the synchronization of infobox tables. This distinction allows us to tackle a broader range of synchronization tasks that require structured reasoning, beyond just textual content. Our dataset and approach are multilingual, concentrating on the synchronization of tables across different languages—an aspect not covered by SUMIE. We investigate how data can be aligned and updated across multiple languages, whereas SUMIE does not explore multilingual contexts. SUMIE generates data using LLM-based synthetic pipelines, whereas our dataset is directly sourced from real-world Wikipedia data, offering more diversity.

SituatedQA (Zhang and Choi, 2021) and TEMPLAMA (Dhingra et al., 2022) target open-domain and cloze-style temporal queries. TempTabQA (Gupta et al., 2023), TIQ (Jia et al., 2024), TRAM (Wang and Zhao, 2024), and the BIG-bench project (et. al., 2023) address temporal reasoning over tables and knowledge bases. More recent work (Tan et al., 2023, 2024) investigates temporal reasoning in large language models (LLMs) using unstructured and synthetic data.

However, none of this work focuses on editing multilingual tables. Some studies focus on Wikipedia-based document editing (Lange et al., 2010; Sáez and Hogan, 2018; Sultana et al., 2012), but not tables. Others apply editing strategies to technical, scientific, legal, and medical tables (Wang et al., 2013; Gottschalk and Demidova, 2017). Expanding our approach to include social, economic, and cultural aspects in table updates would be a valuable direction for future research.

8 Conclusion and Future Work

In this paper, we explored the application of large language models (LLMs) for multilingual information synchronization, focusing on improving the accuracy and coherence of updates to Wikipedia tables in low-resource languages. Our task decomposition strategy significantly outperformed baseline methods, especially in information updating and addition. The Information Updation dataset enabled a more precise evaluation of LLM capabilities. Overall, our findings highlight the potential of LLMs for dynamic data enrichment across diverse architectures, advancing multilingual and low-resource information systems.

Future research could explore several key directions: (a) extending the dataset to include diverse languages and more complex information structures to test LLM generalizability, (b) integrating LLMs with rule-based methods or knowledge graphs for improved factual accuracy, (c) enhancing the model's performance in deletion tasks without weakening its strength in addition and updating, and (d) investigating efficient prompting strategies and fine-tuning techniques to improve scalability and real-world applicability across different model architectures.

Limitations

Even though our research demonstrates significant improvements in multilingual information synchronization using large language models (LLMs), several limitations remain. The performance of the models is highly dependent on the quality and diversity of the pre-training data, which may not fully capture the nuances of low-resource languages, leading to inconsistencies across different linguistic contexts, and across different LLMs. Additionally, although our task decomposition strategy improves performance in information updating and addition tasks, it also increases the number of prompts, resulting in a slight rise in deletion errors. This highlights the need for further refinement to balance the model's strengths in information addition and correction with its ability to manage deletions effectively. The use of closed-source models such as GPT-3.5 and Gemini 1.5 Flash Pro also limits transparency and replicability, while open-source models such as LLAMA 3.0 offer more flexibility but may not achieve the same performance levels. Lastly, the computational demands of our approach, though manageable within our project, could pose challenges for broader scalability, particularly in resource-constrained environments. Future research should focus on developing more efficient and scalable solutions to address these limitations and ensure generalizability across diverse languages and domains.

Ethics Statement

This research on leveraging large language models (LLMs) for multilingual information synchronization involves several ethical considerations. First, there is a risk of reinforcing biases, particularly in low-resource languages where training data may be limited and skewed, potentially leading to the spread of cultural or factual inaccuracies. Ensuring transparency and incorporating mechanisms for human oversight are essential to prevent misinformation, especially when automating updates for public knowledge sources such as Wikipedia. Additionally, respecting intellectual property and data rights is critical when utilizing publicly available datasets, as unauthorized use could raise ethical and legal concerns. The computational cost of training and deploying LLMs also contributes to environmental impacts, highlighting the importance of developing more energy-efficient models. Although this research demonstrates the potential of LLMs to improve information synchronization, addressing these ethical issues is key to responsible and equitable deployment in real-world applications.

Acknowledgements

Research was sponsored by the Army Research Office and was accomplished under Grant Number W911NF-20-1-0080. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. This work was partially funded by ONR Contract N00014-23-1-2364. We extend our gratitude to the annotators who verified our data. We extend our gratitude to the annotators who verified our data extraction and corresponding question answer pairs. We also thank Kashish Ranjan from IIT Guwahati for initial preliminary evaluations of the problem. We extend our sincere appreciation to Jennifer Sheffield from the University of Pennsylvania for her administrative support. Lastly, we extend our appreciation to the reviewing team for their insightful comments.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Eytan Adar, Michael Skinner, and Daniel S. Weld. 2009. Information arbitrage across multi-lingual wikipedia. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, page 94–103, New York, NY, USA. Association for Computing Machinery.

AI@Meta. 2024. Llama 3 model card.

Patti Bao, Brent Hecht, Samuel Carton, Mahmood Quaderi, Michael Horn, and Darren Gergle. 2012. Omnipedia: Bridging the wikipedia language gap. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, page 1075–1084, New York, NY, USA. Association for Computing Machinery.

Gosse Bouma, Sergio Duarte, and Zahurul Islam. 2009. Cross-lingual alignment and completion of Wikipedia templates. In *Proceedings of the Third International Workshop on Cross Lingual Information*

Access: Addressing the Information Need of Multilingual Societies (CLIAWS3), pages 21–29, Boulder, Colorado. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Wenhu Chen, Xinyi Wang, and William Yang Wang. 2021. A dataset for answering time-sensitive questions. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15607–15631, Toronto, Canada. Association for Computational Linguistics.

Christopher Davis, Andrew Caines, Øistein Andersen, Shiva Taslimipoor, Helen Yannakoudakis, Zheng Yuan, Christopher Bryant, Marek Rei, and Paula Buttery. 2024. Prompting open-source and commercial language models for grammatical error correction of english learner text. *arXiv preprint arXiv:2401.07702*.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. 2022. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273.

Aarohi Srivastava et. al. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*.

Simon Gottschalk and Elena Demidova. 2017. Multiwiki: Interlingual text passage alignment in wikipedia. *ACM Trans. Web*, 11(1).

Vivek Gupta, Pranshu Kandoi, Mahek Vora, Shuo Zhang, Yujie He, Ridho Reinanda, and Vivek Srikumar. 2023. TempTabQA: Temporal question answering for semi-structured tables. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2431–2453, Singapore. Association for Computational Linguistics.

Eunjeong Hwang, Yichao Zhou, Beliz Gunel, James Bradley Wendt, and Sandeep Tata. 2024. Sumie: A synthetic benchmark for incremental entity summarization. *arXiv preprint arXiv:2406.05079*.

Robert Iv, Alexandre Passos, Sameer Singh, and Ming-Wei Chang. 2022. FRUIT: Faithfully reflecting updated information in text. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3670–3686, Seattle, United States. Association for Computational Linguistics.

Saemi Jang, Mun Yong Yi, et al. 2016. Utilization of dbpedia mapping in cross lingual wikipedia infobox completion. In *Australasian Joint Conference on Artificial Intelligence*, pages 303–316. Springer.

Zhen Jia, Abdalghani Abujabal, Rishiraj Saha Roy, Jannik Strötgen, and Gerhard Weikum. 2018. Tempquestions: A benchmark for temporal question answering. In *Companion Proceedings of the The Web Conference 2018*, WWW '18, page 1057–1062, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Zhen Jia, Philipp Christmann, and Gerhard Weikum. 2024. Tiq: A benchmark for temporal question answering with implicit time constraints. In *Companion Proceedings of the ACM Web Conference 2024*, WWW '24, page 1394–1399, New York, NY, USA. Association for Computing Machinery.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Siddharth Khincha, Chelsi Jain, Vivek Gupta, Tushar Kataria, and Shuo Zhang. 2023. InfoSync: Information synchronization across multilingual semi-structured tables. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 2536–2559, Toronto, Canada. Association for Computational Linguistics.

Tushar Khot, Harsh Trivedi, Matthew Finlayson, Yao Fu, Kyle Richardson, Peter Clark, and Ashish Sabharwal. 2022. Decomposed prompting: A modular approach for solving complex tasks. *arXiv preprint arXiv:2210.02406*.

Dustin Lange, Christoph Böhm, and Felix Naumann. 2010. Extracting structured information from wikipedia articles to populate infoboxes. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, page 1661–1664, New York, NY, USA. Association for Computing Machinery.

Yinghui Li, Shang Qin, Jingheng Ye, Shirong Ma, Yangning Li, Libo Qin, Xuming Hu, Wenhao Jiang, Hai-Tao Zheng, and Philip S Yu. 2024. Rethinking the roles of large language models in chinese grammatical error correction. *arXiv preprint arXiv:2402.11420*.

Renjie Liu, Yanxiang Zhang, Yun Zhu, Haicheng Sun, Yuanbo Zhang, Michael Xuelin Huang, Shanqing Cai, Lei Meng, and Shumin Zhai. 2024. Proofread: Fixes all errors with one tap. *arXiv preprint arXiv:2406.04523*.

Feipeng Ma, Yizhou Zhou, Yueyi Zhang, Siying Wu, Zheyu Zhang, Zilong He, Fengyun Rao, and Xiaoyan Sun. 2024. Task navigator: Decomposing complex tasks for multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2248–2257.

Sumit Neelam, Udit Sharma, Hima Karanam, Shajith Ikbal, Pavan Kapanipathi, Ibrahim Abdelaziz, Nandana Mihindukulasooriya, Young-Suk Lee, Santosh Srivastava, Cezar Pendus, Saswati Dana, Dinesh Garg, Achille Fokoue, G P Shrivatsa Bhargav, Dinesh Khandelwal, Srinivas Ravishankar, Sairam Gurajada, Maria Chang, Rosario Uceda-Sosa, Salim Roukos, Alexander Gray, Guilherme Lima, Ryan Riegel, Francois Luus, and L V Subramaniam. 2022. SYGMA: A system for generalizable and modular question answering over knowledge bases. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3866–3879, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Nhu Nguyen, Dung Cao, and Anh Nguyen. 2018. Automatically mapping wikipedia infobox attributes to dbpedia properties for fast deployment of vietnamese dbpedia chapter. In *Asian Conference on Intelligent Information and Database Systems*, pages 127–136. Springer.

Thanh Nguyen, Viviane Moreira, Huong Nguyen, Hoa Nguyen, and Juliana Freire. 2011. Multilingual schema matching for wikipedia infoboxes. *Proceedings of the VLDB Endowment*, 5(2).

Qiang Ning, Hao Wu, Rujun Han, Nanyun Peng, Matt Gardner, and Dan Roth. 2020. TORQUE: A reading comprehension dataset of temporal ordering questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1158–1172, Online. Association for Computational Linguistics.

Sheena Panthaplackel, Adrian Benton, and Mark Dredze. 2022. Updated headline generation: Creating updated summaries for evolving news stories. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6438–6461, Dublin, Ireland. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Tomás Sáez and Aidan Hogan. 2018. Automatically generating wikipedia info-boxes from wikidata. In *Companion Proceedings of the The Web Conference 2018*, WWW '18, page 1823–1830, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

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

Alexander Spangher, Xiang Ren, Jonathan May, and Nanyun Peng. 2022. NewsEdits: A news article revision dataset and a novel document-level reasoning challenge. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 127–157, Seattle, United States. Association for Computational Linguistics.

Afroza Sultana, Quazi Mainul Hasan, Ashis Kumer Biswas, Soumyava Das, Habibur Rahman, Chris Ding, and Chengkai Li. 2012. Infobox suggestion for wikipedia entities. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, page 2307–2310, New York, NY, USA. Association for Computing Machinery.

Thang Hoang Ta and Chutiporn Anutariya. 2015. A model for enriching multilingual wikipedias using infobox and wikidata property alignment. In *Joint International Semantic Technology Conference*, pages 335–350. Springer.

Qingyu Tan, Hwee Tou Ng, and Lidong Bing. 2023. Towards benchmarking and improving the temporal reasoning capability of large language models.

Qingyu Tan, Hwee Tou Ng, and Lidong Bing. 2024. Towards robust temporal reasoning of large language models via a multi-hop qa dataset and pseudo-instruction tuning.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Liyuan Wang, Jingyi Xie, Xingxing Zhang, Mingyi Huang, Hang Su, and Jun Zhu. 2024. Hierarchical decomposition of prompt-based continual learning: Rethinking obscured sub-optimality. *Advances in Neural Information Processing Systems*, 36.

Yuqing Wang and Yun Zhao. 2024. TRAM: Benchmarking temporal reasoning for large language models.

Zhigang Wang, Zhixing Li, Juanzi Li, Jie Tang, and Jeff Z. Pan. 2013. Transfer learning based cross-lingual knowledge extraction for Wikipedia. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 641–650, Sofia, Bulgaria. Association for Computational Linguistics.

Michael Zhang and Eunsol Choi. 2021. SituatedQA: Incorporating extra-linguistic contexts into QA. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7371–7387, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Shuo Zhang and Krisztian Balog. 2020. Web table extraction, retrieval, and augmentation: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(2):1–35.

Shuo Zhang, Krisztian Balog, and Jamie Callan. 2020a. Generating categories for sets of entities. CIKM '20, page 1833–1842, New York, NY, USA. Association for Computing Machinery.

Shuo Zhang, Edgar Meij, Krisztian Balog, and Ridho Reinanda. 2020b. Novel entity discovery from web tables. In *Proceedings of The Web Conference 2020*, WWW '20, pages 1298–1308.

Yan Zhang, Thomas Paradis, Lei Hou, Juanzi Li, Jing Zhang, and Haitao Zheng. 2017. Cross-lingual infobox alignment in wikipedia using entity-attribute factor graph. In *International Semantic Web Conference*, pages 745–760. Springer.

A Appendix

A.1 Information Updation Dataset Statistics

Table 4 shows the statistics for the proposed information update dataset grouped by different categories and different languages in the table. The dataset is skewed toward high-resource languages because updated tables on entities of interest are seldom available in low-resource languages such as Afrikaans and Cebuano.

A.2 Metrics Definitions for Example in the main paper

Table 5 shows the tri-align, bi-align and un-aligned set of rows for the example shown in Figure 2 using formal defination from Table 6.

Table 7 shows the metrics defined for each alignment type defined in Section for example presented in Figure 2. The figure illustrates the calculation of various metrics by presenting an example and evaluating each defined metric step by step.

Language	Tables	Category	Tables
af	7	Album	76
ar	120	Athlete	70
ceb	4	City	108
de	105	College	112
en	206	Company	148
es	23	Country	122
fr	123	Musician	138
hi	64	Person	108
ko	93	Stadium	66
nl	21		
ru	131		
sv	15		
tr	18		
zh	18		

Table 4: **Dataset Statistics**. Number of pair of (old,new) tables in the dataset grouped the language and different categories.

Component	Definition	Example Associations
Tri-Align (tr)	Alignment across **Input**, **Gold**, and **Output**.	$\texttt{ir}_1 \leftrightarrow \texttt{gr}_1 \leftrightarrow \texttt{or}_1$ $\texttt{ir}_2 \leftrightarrow \texttt{gr}_2 \leftrightarrow \texttt{or}_2$
Bi-Align (bi)	Alignment between only two components.	**Input** ↔ **Gold:** $\texttt{ir}_3 \leftrightarrow \texttt{gr}_3$ **Gold** ↔ **Output:** $\texttt{gr}_4 \leftrightarrow \texttt{or}_3$, $\texttt{gr}_7 \leftrightarrow \texttt{or}_5$
Un-Aligned (un)	Elements not aligned with any component.	**Input:** $\texttt{ir}_4$ **Gold:** $\texttt{gr}_5, \texttt{gr}_6, \texttt{gr}_8$ **Output:** $\texttt{or}_4, \texttt{or}_6$

Table 5: Summary of Alignment Components and Example Associations for the example in Figure 2.

Type	Formal Definition
	Trialign
All	$\{(i, o, g) \mid i \in \text{Ti}, o \in \text{To}, g \in \text{Tg}, \text{Aligned}(i, g) \wedge \text{Aligned}(o, g)\}$
	Bialign
Input	$\{i \in \text{Ti} \mid \exists g \in \text{Tg}, \forall o \in \text{To}, \text{Aligned}(i, g) \wedge \neg\text{Aligned}(o, g)\}$
Output	$\{o \in \text{To} \mid \exists g \in \text{Tg}, \forall i \in \text{Ti}, \text{Aligned}(o, g) \wedge \neg\text{Aligned}(i, g)\}$
	UnAlign
Input	$\{i \in \text{Ti} \mid \forall g \in \text{Tg}, \neg\text{Aligned}(i, g)\}$
Output	$\{o \in \text{To} \mid \forall g \in \text{Tg}, \neg\text{Aligned}(o, g)\}$
Gold	$\{g \in \text{Tg} \mid \forall i \in \text{Ti}, \forall o \in \text{To}, \neg\text{Aligned}(i, g) \wedge \neg\text{Aligned}(o, g)\}$

Table 6: Definitions of alignment groups

A.3 Results with Single Model Evaluations

In the main paper (Table 1), we reported results based on voting across multiple model predictions. However, we also evaluated the performance using individual models without voting. These results are presented in Table 11. It is clearly demon-

Component	Notation	Information Metrics
Tri-Align (tr)	$\mathrm{tr}(o, g)$: Gold elements aligned with Output. $\mathrm{tr}(i, g)$: Gold elements aligned with Input. $\lvert g \rvert$: Total Gold elements.	**Information Updated:** $\frac{\mathrm{tr}(o,g)-\mathrm{tr}(i,g)}{\lvert g \rvert} = \frac{2}{8}$ Represents the net addition of relevant information to the Output.
Bi-Align (bi)	$\mathrm{bi}(o, g)$: Gold elements aligned with Output but not Input. $\mathrm{bi}(i, g)$: Gold elements aligned with Input but not Output. $\lvert g \rvert$: Total Gold elements.	**Noisy Information Added:** $\frac{\mathrm{bi}(o,g)}{\lvert g \rvert} = \frac{2}{8}$ Represents misaligned additions from the Output. **Noisy Information Deleted:** $\frac{\mathrm{bi}(i,g)}{\lvert g \rvert} = \frac{1}{8}$ Represents omissions from the Input that were relevant.
Un-Aligned (un)	$\mathrm{un}(g)$: Unaligned Gold elements. $\mathrm{un}(i)$: Unaligned Input elements. $\mathrm{un}(o)$: Unaligned Output elements. $\lvert g \rvert$: Total Gold elements. $\lvert i \rvert$: Total Input elements. $\lvert o \rvert$: Total Output elements.	**Missing Information (Gold):** $\frac{\mathrm{un}(g)}{\lvert g \rvert} = \frac{3}{8}$ Represents the proportion of relevant Gold elements left unaligned. **Noisy Information Added (Input):** $\frac{\mathrm{un}(i)}{\lvert i \rvert} = \frac{1}{4}$ Indicates the proportion of irrelevant elements in Input. **Noisy Information Added (Output):** $\frac{\mathrm{un}(o)}{\lvert o \rvert} = \frac{2}{6}$ Indicates the proportion of irrelevant elements in Output.

Table 7: Summary of Notation and Metrics for Alignment Components with Information Details

strated here that our proposed approach outperforms all other approaches discussed consistently across models, further proving its efficacy. From the table we can clearly see that Gemini is the best performing model across most metrics.

The prompts used to generate outputs are shown in B. The evaluation prompt used is shown in B.5(it is paired with several examples across multiple languages annotated by us covering a variety of evaluation examples to ensure efficacy).

keys	values
Name	Albert Einstein
Birth date	March 14, 1879
Profession	Theoretical Physicist
Country	Germany, United States

Table 8: **Current Table** in English.

keys	values
Name	Albert Einstein
Birth date	14 March 1879
Profession	Physicist
Country	Germany

Table 9: **Outdated Table** which Needs to be updated, translated to English form German.

A.4 Importance of Using Knowledge Graphs.

Our method involves table merging, which often faces challenges due to variations in naming conventions, relationships, and structures across different data sources (in our case table form multilingual pages). To address these issues, the Hierarchical Task Decomposition Prompt utilizes knowledge graphs (KGs), providing a unified structured, hierarchical representation of the data. This structure enables more effective reasoning and merging. For example, let's consider two tables Albert Einstein, current table 8 and outdated table that needs to be updated Table 9.

By converting these tables into knowledge graphs, we can align: "Birthdate" and "Date of Birth" as the same entity and "Profession" and "Occupation" as related attributes very easily and accurately. This knowledge graphs representation simplifies merging for the LLM:

- **Handling Variations in Data Representation**, such as "Birthdate" vs. "Date of Birth" or "Profession" vs. "Occupation". Directly using LLMs on these tables may cause confusion or difficulty in aligning these terms. The LLM might not explicitly recognize that these attributes refer to the same entity. By converting the tables into knowledge graphs, we explicitly capture the relationships between the entities. For example, in a knowledge graph, the entity "Albert Einstein" is connected to "Birthdate", "Profession", and "Country" with clear edges that denote these relationships. Even if different names are used (like "Date of Birth" and "Birthdate"), the LLM can recognize that these refer to the same concept by examining the structure and context in the KG.

- **Improved Alignment and Merging.** With KGs, the LLM can easily align data across tables based on the semantic relationships represented in the graph. For example: "Birthdate" in Table 1 and "Date of Birth" in Table 2 refer to the same information. "Profession" in Table 1 and "Occupation" in Table 2 are related attributes. Similarly, "Country" and "Nationality" refer to the same concept.

  With this graph-based representation, merging the two tables becomes much more straightforward. The graph structure helps resolve ambiguities between different terminologies and aligns the data correctly. The LLM can leverage the hierarchical relationships (e.g., Person → Birthdate → 14 March 1879) to merge the two infoboxes into a unified representation. After converting both tables into knowledge graphs and resolving the semantic mappings, the merged table would look Table 10.

keys	values
Person	Albert Einstein
Birthdate	14 March 1879
Profession/Occupation	Theoretical Physicist
Birth Country	Germany
Nationality	Germany, United States

Table 10: **Merged Table.**

- **Improved Reasoning with LLM.** The knowledge graph approach improves performance over directly using LLMs on raw tables for the following reasons:
  - *Hierarchical Reasoning*. The hierarchical nature of KGs enables the LLM to reason more effectively about the relationships between entities and their attributes. This is particularly useful in complex tasks like table merging, where identifying relationships between entities in different tables is crucial.
  - *Merging Benefit Reasoning.* With KGs, merging data becomes more straightforward because the relationships between entities are explicitly defined. The LLM can merge information by focusing on the nodes and edges that connect related concepts, leading to more accurate integration of data and better reasoning.

Converting tables into knowledge graphs allows the LLM to reason effectively over hierarchical and relational data, handling variations in data representation with greater precision. This approach simplifies tasks like table merging, enabling the LLM to align data, resolve ambiguities, and generate more accurate merged results.

	Trialign Rows (Tr)	Bialign Rows (Bi)	UnAlign Gold (UInput BiAlign (Bi)		
Methods	Updated ↑	Added (%) ↑	Added (#Rows) ↑	Missed (G) ↓	Delete (I) ↓
InfoSync (Khincha et al., 2023)	0.94	12.18	2.99	4.67	0.35
GPT 3.5					
Direct Prompt	1.34	5.44	5.14	4.03	0.43
Align-Update (Two Prompts)	-0.37	4.45	0.88	**3.39**	**0.29**
Align-Update (Joint Prompt)	-0.64	1.08	0.75	5.03	0.64
Our Proposed Decomposition Approach					
Direct Decompose Prompt	0.65	5.8	1.75	5.76	0.78
Translation(+BackTrans)	0.31	5.67	1.85	5.67	1.04
+Merge and Alignment	0.42	8.75	2.66	4.84	1.13
+Knowledge Graph	**0.76**	**12.32**	**3.53**	3.8	1.06
Gemini 1.5 Flash Pro					
Direct Prompt	1.27	17.29	5.12	4.02	0.42
Align-Update (Two Prompts)	-0.97	17.24	4.07	3.44	**0.07**
Align-Update (Joint Prompt)	1.14	20.36	4.83	3.11	0.11
Our Proposed Decomposition Approach					
Direct Decompose Prompt	1.04	15.67	3.63	4.03	0.12
Translation(+BackTrans)	0.59	23	5.24	2.67	0.05
+Merge and Alignment	1.77	22.84	**6.19**	**1.91**	0.16
+Knowledge Graph	**2.23**	**25.22**	5.6	2.09	0.12
LLAMA 3.0 (70B)					
Direct Prompt	1.25	16.27	5.01	4.14	0.42
Align-Update (Two Prompts)	-0.97	16.09	3.99	3.53	**0.06**
Align-Update (Joint Prompt)	1.04	19.29	4.75	3.22	0.12
Our Proposed Decomposition Approach					
Direct Decompose Prompt	1	14.72	3.55	4.15	0.13
Translation(+BackTrans)	0.96	21.96	5.16	2.79	0.05
+Merge and Alignment	1.81	21.83	6.11	2.01	0.16
+Knowledge Graph	**2.38**	**24.2**	**5.52**	**2.18**	0.17
Human (100 examples)	1.75	21.44	5.6	2.09	0.12

Table 11: **Information Updation Results for individual LLMs.** GPT3.5, Gemini 1.5 Flash Pro, LLAMA 3.0 (70B) individual performance.

B Prompt Examples

This section presents example prompts used for hierarchical task decomposition and evaluation in our experiments.

B.1 Hierarchical Decomposition Prompt

B.1.1 Translation(x -> English)

Translate the following of Category CATEGORY into English, and provide only the translated table as the output. Ensure that strings with apostrophes are escaped properly using a backslash. Output table Schema: [["key","value"], ["key","value"]]
Table:

B.1.2 Table to Knowledge Graph Conversion

Please convert the following table into a knowledge graph and provide the final knowledge graph in a structured json format. This table is from the category The Output should be in a nested dictionary format. Ensure you do not miss any information in the original table.
Example Output Knowledge Graph: { "Person": { "Name": "Karla Camila Cabello Estrabao", "Born": "March 3, 1997", "Age": "24", "Birthplace": "Cojímar, Havana, Cuba" }, "Occupation": { "Primary": "Singer", "Additional": ["Songwriter", "Actress"] } ....}

B.1.3 Knowledge Graph Merge or Alignments

> Given two knowledge graphs containing information about an entity, your task is to merge the graphs while adhering to the following guidelines:
> Avoid Duplicate Entries: Ensure that there are no duplicate nodes and relations in the merged knowledge graph.
> Resolve Conflicting Information: In cases where there is conflicting information for a specific node, use the most updated value to resolve the conflict. If you are still not able to merge the conflict, then prefer the value in Graph B. When you resolve a conflict, only one of the rows should finally be outputted, not both.
> Merge Redundant Rows: Explicitly check for and merge redundant rows holding the same information. Combine them into a single entry, and only one of them should be outputted.
> Ultimate Goal: Create a merged knowledge graph that includes the latest and most accurate information available, without any missing entries. Do not remove any entry during the merging process. Provide only the merged knowledge graph as the output.
> Knowledge graphs:

B.1.4 Back-Translation

> Convert the knowledge graph into an entity centric table in the format of a list of lists.
> Here is an example conversion of knowledge graph A to table A:
> Graph A:
> Table A:
> Now convert Knowledge Graph G to table G following similar keys to table A:
> Knowledge Graph G:
> Ensure that strings with apostrophes are escaped properly using a backslash. Output table Schema: [["key","value"], ["key","value"]

B.1.5 Translation(English -> x)

> Translate the following English language table of Category Ensure that strings with apostrophes are escaped properly using a backslash.
> Here is an example translation:
> Original Table:
> Translated Table:
> Now translate the following table:
> Output table Schema: [["key","value"], ["key","value"]]

B.2 Align-Update-Joint

This prompt is used for Align-Update (Joint Prompts). For Align-Update (Two Prompts), we separate the prompts into two parts: one for alignment and one for the update tasks.

> Your task is to update Table A(To help you in the task, you are given a set of alignments. Alignments are a mapping between two tables that match similar information. Use these alignments as a reference to make updates as only aligned rows need to be considered while making updates. Alignments are in the following format: [[’Table A Aligned Key 1’],[’Table B Aligned Key 1’], [’Table A Aligned Key 2’],[’Table B Aligned Key 2’],]
> Follow these steps:
> Identify missing or outdated information in Table A compared to its aligned information in Table B, and update it with the corresponding information from Table B. You should add any missing rows present in Table B that are not present in table A. These would be rows of table B that are not present in the set of alignments. You should also fix any wrong, outdated or missing information present in Table using the set of alignments. Your solution should ensure that Table A contains complete and accurate information about the entity using data from Table B. Table A :
> Table B :
> Alignments:
> Provide the updated Table A only in language Output table Schema: [["key","value"], ["key","value"]]

B.3 Direct Decompose

Your task is to update Table A(
Follow these instructions: 1) Translate both tables to English. 2) Create a merged table combining the information from both tables ENSURING that you fix any wrong, outdated or missing information present in both tables. 3) Use the Merged table to update the translated version of Table A. 4) Translate table A back to
Your solution should ensure that Table A contains complete and accurate information about the entity using data from Table B.
Table A :
Table B :
Provide the updated Table A ONLY in language Ensure that strings with apostrophes are escaped properly using a backslash. Output table Schema: [["key","value"], ["key","value"]]

B.4 Just Alignments Prompt

Please provide a list of aligned keys by matching Table G keys with suitable Table A keys, ensuring that they have similar semantic values. Allow for multi-alignments where appropriate. If no suitable alignment is found, please skip that key. Do not change the way a key is written and use the exact representation while making alignments.
Tables for Alignment(language):
Table A:
Table G:
Output Schema:
[[A-key,G-key], [A-key,G-key]....]

B.5 Evaluation Prompt

Your task involves analyzing two sets of key-value pair tables. Begin by translating the tables to English. Then, extract all pertinent fine-grained details from each table. Then, delve into the semantic content, disregarding minor differences due to formatting, grammar, and language nuances.
Within the tables, information may fall into two categories:
'Similar Information': Information common to both tables

- **'Consistent Information'**: Both tables contain identical data with possible differences in format.
- **'Contradictory Information'**: Tables present conflicting data with clear difference in meaning.

Note: While analyzing the information, especially for similar information from two tables, solely focus on the semantic content, disregarding any minor differences due to formatting, grammar, and linguistic variations. While comparing numerical information, allow a reasonable error percentage that you consider acceptable before presenting information as inconsistent. Allow the same error margin for other types of information such as coordinates. Be lenient in grouping information as 'Consistent' when slight differences still refer to the overall same data.
'Unique Information': Information exclusive to one table.
Your comparison should result in four types of information:

- **Similar and consistent information**: `similar_consistent`
- **Similar and contradictory information**: `similar_contradictory`
- **Table 1 unique information**: `table1_unique`
- **Table 2 unique information**: `table2_unique`

Here are the test tables provided in language :
Table 1:
Table 2:
Note: While comparing information, solely focus on the semantic content, disregarding formatting, grammar, and language nuances.