Multilingual Wikidata-Based Assessment of Knowledge in Large Language Models

Damien Nouvel[1,*,†], Daria Tupikina[1,*,†]

[1]Research team ERTIM of INALCO, 2 rue de Lille, Paris, 75013, France

Abstract

Retrieval-Augmented Generation systems have now become very widespread. While these architectures can significantly reduce factual errors, existing frameworks remain vulnerable to structural hallucinations and are heavily dependent on both prompt and context language. This paper presents an empirical evaluation of Wikidata-based Retrieval-Augmented Generation (RAG) for multilingual factual question answering. We compare the performance of four open-weight large language models from Ollama across four languages (English, French, Russian, and Chinese) and three operational modes: baseline (no RAG), text-based RAG with injected Wikipedia content, and structured tool-calling RAG (Tool-RAG). Our automated pipeline executed 76,800 queries to extract biographical attributes from 400 Wikidata human entities of various citizenships. Evaluating outputs with the token-level BERTScore F_1 metric reveals that accuracy of performance depends fundamentally on the targeted property and its underlying data type, while the impact of prompt and data language was less important. Qwen is the top performer among all four Ollama models, achieving a perfect BERTScore F_1 of 1.0 in RAG and tool-calling modes for certain properties.

Keywords

Retrieval Augmented Generation, Tool-calling, Structured Data, Wikidata, Multilingual Benchmarking, Factual Question Answering

1. Introduction

Retrieval-Augmented Generation (RAG) has become a very popular method for operating AI on a specific knowledge base. However, the reliability of this approach is still uncertain[1] and a number of recent studies published between 2025 and 2026 have assessed ontology-based RAG from various architectural perspectives. These works encompass both context-driven text generation and structured information extraction guided by knowledge graphs [2, 3, 4, 5], alongside the converse task of generating knowledge graphs directly from raw textual data [6].

To date, existing research has been primarily focused on five core analytical axes. The first axis concerns the application of ontology-based RAG to specific domains, such as medicine, law, financial technology, and culinary datasets [2, 7]. The second axis focuses on task-specific implementations, including question answering, question generation, document-level relation extraction [3], text-based ontology construction, dialogue systems, and informative text generation. The third axis addresses system optimization, specifically comparing structural data file formats (e.g., RDF, Turtle, OWL) [8]. The fourth axis concerns contrasting text-based RAG against tool-calling as described by Huang et al[7]. Finally, the last axis is the comparison of various evaluation metrics and benchmarks regarding the RAG systems [9].

Despite these diverse contributions, there remains a critical gap in the literature regarding how these retrieval paradigms behave under multilingual and multicultural conditions. To the best of our knowledge, no existing study has systematically compared the gains of RAG ant tool-calling modes

RAGE-KG2026: The Third International Workshop on Retrieval-Augmented Generation Enabled by Knowledge Graphs, co-located with ISWC 2026, October 25–29, 2026, Bari, Italy

*Corresponding author.

†These authors contributed equally.

damien.nouvel@inalco.fr (D. Nouvel); 45018280@parisnanterre.fr (D. Tupikina)

https://damien.nouvels.net (D. Nouvel); https://github.com/ddt2000fr (D. Tupikina)

0000-0001-8866-4028 (D. Nouvel); 0009-0009-9326-8471 (D. Tupikina)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

published 2026-11-04

CEUR-WS.org/Vol-4312/short12.pdf

applied to different languages. Furthermore, we aim to investigate whether there is an alignment between the prompt language, the data language, and the entities' citizenship. These aspects constitute the primary novelty of our research.

Consequently, this work addresses two main research questions:

- **RQ1**: How much do the language of the prompt and the culture of an entity affect the capacity of AI models to accurately answer factual questions?
- **RQ2**: To what extent does the use of RAG and structured functions (Tool-calling) improve the accuracy and linguistic reliability of LLMs in multilingual context?

Our study addresses these questions by focusing on two primary objectives. The first goal is to evaluate RAG system's robustness in four distinct languages with highly varied syntactic and graphical systems (English, French, Russian, and Chinese). The second purpose of our study is to provide an empirical comparison between text-based RAG and structured tool-calling configurations. We conduct experiments by evaluating four open-weight large language models with varying parameter sizes, across four target samples that are categorized by the entities' citizenship.

2. Related work

The intersection of Retrieval-Augmented Generation (RAG), ontologies, structured data, and function-calling has been the subject of several recent studies aiming to improve Large Language Model (LLM) accuracy for factual information. Focusing on the architectural side of knowledge injection, ontology-based RAG has been shown to significantly reduce error rates in LLMs, though performance remains heavily tied to the specific format of the injected data. For instance, research evaluating small, medium, and large models under zero-shot prompting techniques discovered that while ontology-grounded RAG successfully reduces model error rates, the resulting cosine similarity of the generated answers depends significantly on the underlying ontology file format [8].

The application of ontology-grounded frameworks (such as OG-RAG) has demonstrated broad utility across diverse domains, including healthcare, law, agriculture, journalism, and research, according to Sharma et al. [4]. Evaluations of OG-RAG across four different LLMs in their experiment showed a 55% increase in the recall of accurate facts, a 40% improvement in response correctness, 30% faster response attribution to context, and a 27% boost in fact-based reasoning accuracy compared to baseline methods.

To optimize extraction accuracy, alternative prompting mechanisms have also been widely explored. Research into Competency Question Generation from ontological triplets investigated the use of few-shot prompting techniques with examples, demonstrating that they substantially improve both model precision and recall [10]. However, researchers in that study ultimately questioned whether these observed improvements are worth the additional computational cost associated with ontology injection. In contrast to these few-shot approaches, our benchmark operates strictly using zero-shot prompting techniques across all evaluated modes to isolate the models' baseline capabilities.

Wikidata was already used as a data source and reference for the experiment described by Boudourides [1]. Namely, it was used to evaluate a model's reconstructed knowledge about philosophers. The evaluation involved seven fields: date of birth, date of death, place of birth, country of citizenship, connection to other philosophers, and field of work. While the results demonstrated a high level of hallucinations in the model's responses across all fields, authors showed that the models had more difficulties with set-valued fields containing relational knowledge, such as connections to other philosophers. In our research, we also use date of birth and place of birth to question the models, to which we add sex and occupation, the latter being a set.

Tool calling represents a key advancement in bypassing these context filters to boost attribute extraction accuracy. Recent research by Huang et al. applying tool calling to the complex domain of medical consultation showed that it successfully improves attribute extraction and accuracy of answers for models like Qwen and ChatGPT. We used an analogical tool schema as in this research. This

explicitly tests the model's capacity to navigate structured property extraction by deciding whether or not to invoke the established tool function schema across varying multilingual syntactic structures.

Recent studies have increasingly focused on assessing the multilingual factual capabilities and cultural knowledge of large language models. Benchmarks such as MultiLoKo [11] and ECLeKTic [12] show that LLMs frequently struggle with cross-lingual knowledge transfer, demonstrating factual inconsistencies when the query language differs from the cultural origin of the information. However, these studies do not investigate the comparative impact of different operational modes across diverse linguistic contexts, a gap this paper directly addresses.

Several studies evaluate RAG performance in a multilingual context. For example, the study of Chirkova et al. [5] presents an experiment with RAG across 13 languages. The authors also employ open questions to evaluate the models' knowledge. They highlight the importance of writing prompts in the same language as both the input database and the expected answer.

3. Methodology

3.1. Data Extraction and Preprocessing

We downloaded the latest Wikidata JSON dump of all Wikipedia entities, dated May 16, 2026. Then we developed a script to extract only human entities that have Wikipedia pages in all four target languages: English, French, Russian, and Chinese.

More precisely, for every human entity we extracted the Wikidata ID, entity label, description, and Wikipedia page title, as well as five specific properties across all four languages: `date_of_birth`, `place_of_birth`, `sex_or_gender`, `citizenship`, `occupation`.

We then simplified Wikidata's original data structure and saved the output in JSON format. We then filtered the dataset to remove human entities containing a `null` value resulting from missing properties in any of the four target languages. The resulting JSON file structure is shown in Table 1.

Table 1
JSON data structure

Field	Data type	Role
ID	String with entity number, for example, Q42	Unique Wikidata entity ID
Label	Text string	Entity's name
Description	Text string	Short entity description from Wikidata
Wikipedia page title	Text string	Corresponding Wikipedia page name
Date of birth	Date in DD-MM-YYYY format	Entity's date of birth property
Place of birth	Text string	Entity's place of birth property
Sex or gender	Text string	Entity's sex or gender property
Citizenship	Text string	Entity's citizenship property
Occupation	List of text strings	Entity's occupation property

3.2. Samples Preparation

Following the preprocessing phase, we prepared four distinct samples based on the entities' citizenship, with each sample containing 100 random entities that have the following citizenship property:

- Great Britain or USA
- France
- Russian Empire, USSR, or Russian Federation
- Qing Dynasty, Republic of China, or People's Republic of China

For each entity, we injected the corresponding Wikipedia page text, pre-cleaned using the Beautiful-Soup library.

3.3. Prompts and LLM queries

We developed the main pipeline that makes requests to four Ollama [13] models : `llama3.2:latest`, `granite3.3`, `qwen3:4b` and `mistral-nemo:latest` (see Table 2 for details).

This pipeline and evaluation scripts were executed locally on an NVIDIA GeForce RTX 3090 GPU equipped with 24 GB of VRAM.

Table 2
Models' characteristics

Characteristics	`llama3.2:latest`	`granite3.3`	`qwen3:4b`	`mistral-nemo:latest`
Number of parameters	3.2B	8.2B	4.0B	7.2B
Size	2.0 GB	4.9 GB	2.5 GB	4.4 GB

Each model is tested in three operational modes :

- **no rag**
  The LLM responds solely based on its internal knowledge. This mode serves as a baseline and measures what the model already knows.
- **rag**
  The HTML content of the Wikidata page is cleaned using BeautifulSoup and then injected into the prompt as context. The prompt explicitly requires using only this context.
- **tool rag**
  The LLM has access to a `get_field` tool declared via Ollama's native tool calling API. The entity is pre-loaded on the code side; the tool returns the structured value of the requested field. The LLM must call the tool and then formulate its response. A loop of up to four steps frames the conversation, and a tool_not_called status signals cases where the model ignores the tool.

For each mode, we prepared prompts in all four languages : English, French, Russian and Chinese. Prompt versions in English for all three modes as well as the tool schema for tool calling mode are shown in Appendix A.

4. Evaluation

We evaluated models' responses using the BERTScore `F1` metric [14]. This metric tokenizes both the model's hypothesis and the reference text, calculates the cosine similarity between every token pair across the two sequences, and computes a final `F1 score`. We determined that this metric is appropriate for our experiment because several targeted properties inherently require answers in the form of a list of words. For instance, the `place_of_birth` attribute can vary significantly in length because a model's answer may optionally mention the country, region, and specific district. Similarly, the occupation property often comprises an unpredictable list of multiple professions, titles, and roles. For this reason, it is particularly important that the evaluation metric accounts for each token individually, in contrast to dense embedding-based metrics, such as BGE, which map the entire response into a single aggregate vector.

To evaluate the models' answers against a ground-truth baseline, we used Wikidata as our reference source. For each evaluated property, we extracted the content of the corresponding field directly from the preprocessed .ndjson file to serve as the reference value.

5. Results

Table 3 reports mean BERTScore F_1 of the four models by citizenship. Qwen emerged as the most performant model across all evaluated samples in this experiment. According to the same table, the

uniform mean BERTScore F1 across citizenship samples confirms that our dataset is well-balanced across different cultures.

When comparing models' performance on different citizenship samples, Llama obtained the highest performance on the Russian sample, Granite on the French one, Qwen on the Chinese one, and Mistral on the the Russian one. However, this performance variation is very moderate. We conclude that there is no strong correlation between models accuracy and entity citizenship, as all models show international knowledge of humans.

Table 3
Models' mean BERTScore F_1 by citizenship sample (aggregated across all modes)

Model	Eng samp	Fr samp	Ru samp	Zh samp	Mean by model
llama	0.725	0.731	0.735	0.728	0.730
granite	0.715	0.717	0.716	0.713	0.716
qwen	**0.835**	**0.843**	**0.845**	**0.858**	**0.845**
mistral	0.770	0.768	0.784	0.775	0.774
Mean by sample	0.761	0.765	0.770	0.768	0.766

In Table 4, we report mean BERTScore F_1 for languages of prompts and data provided to RAG by citizenship sample. As other studies already stated, there is a clear domination of the English language when prompting models.

Additionally, the experimental results reveal that these models are more efficient when both the underlying data and the prompt are in the English language. This trend holds true even when the entity is of non-English-speaking citizenship, despite the assumption that fewer English-language texts exist for such individuals compared to resources in their native languages.

Table 4
Mean BERTScore F_1 by language and citizenship sample (aggregated across all models and modes)

Language	Eng samp	Fr samp	Ru samp	Zh samp	Mean by language
English	**0.805**	**0.805**	**0.804**	**0.800**	**0.804**
French	0.709	0.715	0.717	0.712	0.728
Russian	0.721	0.721	0.734	0.721	0.739
Chinese	0.765	0.774	0.779	0.794	0.794

As for the properties, Table 5 in Appendix B demonstrates that the highest performance is observed for `sex_or_gender` and `date_of_birth`. The most difficult field to extract is the `occupation` property.

Concerning three modes, Table 5 shows that the best results were achieved in the **rag** mode through the injection of raw Wikipedia HTML pages.

According to the Figure 1,a notable pattern can be observed across all evaluated models, where accuracy of performance depends directly on the specific property being extracted. For instance, the Llama model is most efficient in **no rag** mode when answering questions about `date_of_birth` , in rag mode for `place_of_birth` and `sex_or_gender`, and in **tool rag** mode for the `occupation` property. The Qwen model proves most efficient in **tool rag** mode for all properties except occupation, for which standard **rag** mode gets higher accuracy. Granite demonstrates its best results in **rag** mode for the `place_of_birth` and occupation fields, but shifts to being more effective in **tool rag** mode for `date_of_birth` and `sex_or_gender`. Lastly, Mistral shows the highest performance in rag mode for all targeted attributes except `date_of_birth`, which evaluates better in the baseline **no rag** configuration.

Overall, the best results were achieved by Qwen, attaining a perfect BERTScore F_1 of 1.0 (100%). Table 6 in Appendix B shows the details of the four configurations that achieved this result. Specifically, for the `date_of_birth` property, this result was obtained in **tool rag** mode across all four citizenship samples using Chinese as the prompt language. The same model and language combination also yielded

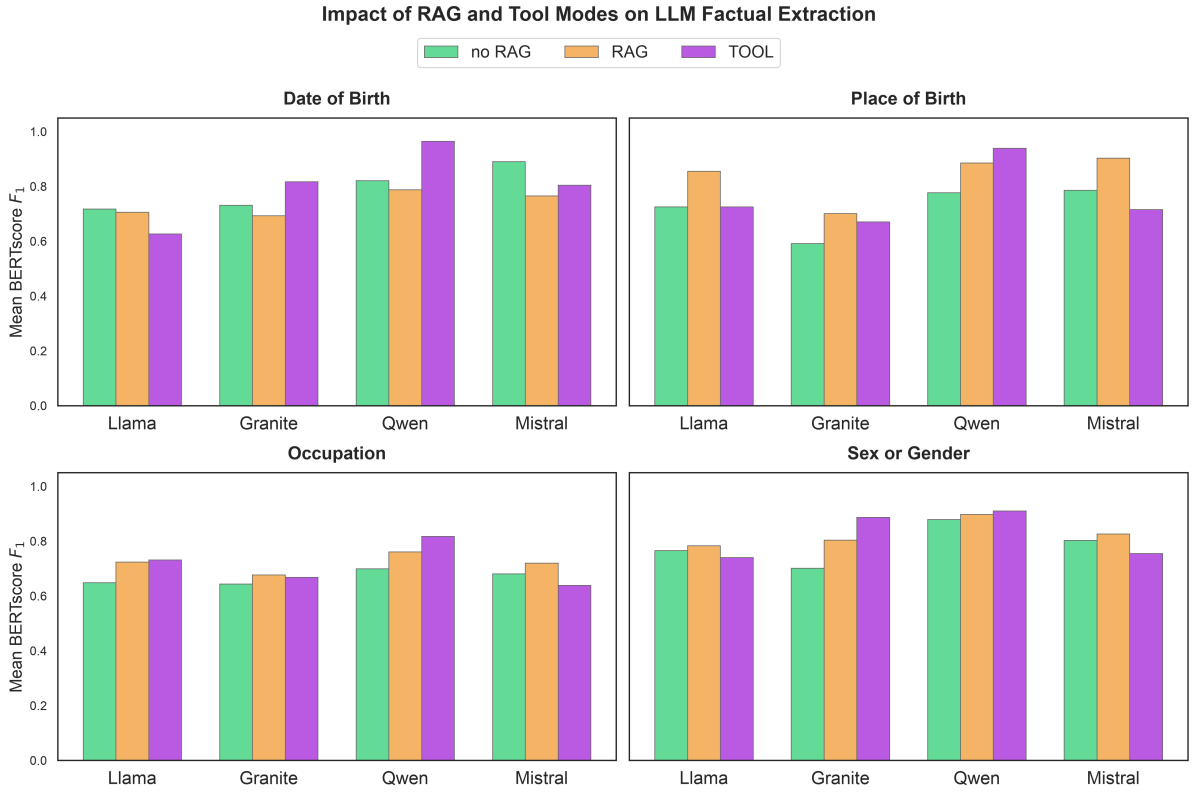


Figure 1: Mean BERTScore F1 by models and modes for the four properties.

the best results for the `place_of_birth` property in **tool rag** mode for the French citizenship dataset. Furthermore, Qwen achieved peak performance for the `sex_or_gender` property across all citizenship samples using both the **rag** and **tool rag** modes.

6. Conclusion

In this work, we extracted 400 Wikidata human entities along with some of their properties, prepared four distinct samples, developed language-specific prompts and the query script, conducted 76,800 model queries, and evaluated the outputs using the BERTScore F_1 metric.

Our evaluation demonstrates two primary insights. Firstly, although language selection plays a critical role in system performance, there is no linguistic and cultural alignment, when processing a language that corresponds directly to the target entity's citizenship. English emerged as the most effective language for prompting, whereas French was consistently the least effective, a trend particularly pronounced within the **tool rag** configuration. Entity citizenship in general did not have a statistically significant influence on overall models' performance, as all four citizenship samples achieved highly comparable mean BERTScore F_1.

Secondly, rag and tool rag modes systematically improve models' performance, with 0.039 and 0.035 mean gain respectively. However, accuracy of performance in a specific mode depends heavily on the prompt language, as mentioned above, and on target property and its data type. For instance, the standard text-based **rag** mode demonstrated the highest BERTScore F_1 gains for the `place_of_birth` property but resulted in the lowest (and occasionally negative) gains for `date_of_birth`. Conversely, the **tool rag** configuration showed its lowest performance for `date_of_birth` but achieved its highest overall accuracy gains when extracting `occupation`.

This experiment provides us with a comprehensive overview of parameters that affect the methods to extract knowledge from models with RAG capabilities. However, our conclusions rely on evaluations based on BERTScore F1 and should be extended to other metrics and human judgement.

Acknowledgments

The authors would like to thank INALCO students Damien Biguet, Mélanie Zhuge, Yongcan Liu, Maiwenn Plevenage, Yifan Miao and Constance Sainte Luce for their valuable contributions to this research, particularly for their assistance with script development and data preprocessing. We also express our gratitude to Youjia Zeng for her help with the translation of the prompts in Chinese.

Declaration on Generative AI

During the preparation of this work, the authors used Gemini to improve the phrasing, layout, spelling, and grammar of this article. After using this service, the authors reviewed and edited the content as needed and take full responsibility for the publication's final content.

References

[1] M. Boudourides, Structural hallucination in large language models: A network-based evaluation of knowledge organization and citation integrity, arXiv preprint arXiv:2603.01341 (2026).

[2] B. Gajderowicz, A. Bhardwaj, M. Fox, Rag and ontologies for information retrieval: A., in: Convergence of large-language models and ontologies (ONTOLLM): Proceedings of the joint ontology workshops (JOWO)—episode XI: The sicilian summer under the etna, 2025.

[3] W. J. Schmidt, D. Rincon-Yanez, I. Grangel-González, A. Paschke, E. Kharlamov, Document-level relation extraction with ontology-guided rag, in: Linking Meaning: Semantic Technologies Shaping the Future of AI, IOS Press, 2025, pp. 19–27.

[4] K. Sharma, P. Kumar, Y. Li, Og-rag: ontology-grounded retrieval-augmented generation for large language models, in: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, 2025, pp. 32950–32969.

[5] N. Chirkova, D. Rau, H. Déjean, T. Formal, S. Clinchant, V. Nikoulina, Retrieval-augmented generation in multilingual settings, in: Proceedings of the 1st Workshop on Towards Knowledgeable Language Models (KnowLLM 2024), 2024, pp. 177–188.

[6] W. Baldwin, S. Ghanavati, Knowledge graph representations for llm-based policy compliance reasoning, arXiv preprint arXiv:2604.27713 (2026).

[7] Z. Huang, K. Xue, Y. Fan, L. Mu, R. Liu, T. Ruan, S. Zhang, X. Zhang, Tool calling: Enhancing medication consultation via retrieval-augmented large language models, arXiv preprint arXiv:2404.17897 (2024).

[8] H. B. Giglou, S. Burbach, F. Compagno, M. Ismail, G. Singh, S. Rudolph, Beyond statistical parroting: Hard-coding truth into llms via ontologies (2025).

[9] L. Guder, J. P. de Moura, A. Accorsi, G. L. d. Amaral, M. C. Magnaguagno, F. Meneguzzi, M. S. Pinho, D. Griebler, Rage: A retrieval-augmented generation evaluation framework, arXiv preprint arXiv:2605.27445 (2026).

[10] M. Li, R. Alharbi, J. de Berardinis, V. Tamma, T. R. Payne, Retrofit-cq revisited: Leveraging ontology triples and context in few-shot llm prompting, ISWC, 2025.

[11] D. Hupkes, N. Bogoychev, Multiloko: a multilingual local knowledge benchmark for llms spanning 31 languages, arXiv preprint arXiv:2504.10356 (2025).

[12] O. Goldman, U. Shaham, D. Malkin, S. Eiger, A. Hassidim, Y. Matias, J. Maynez, A. M. Gilady, J. Riesa, S. Rijhwani, et al., Eclektic: a novel challenge set for evaluation of cross-lingual knowledge transfer, arXiv preprint arXiv:2502.21228 (2025).

[13] Ollama Team, Ollama: Run llama 3, mistral, and other large language models locally, 2024. URL: https://github.com/ollama/ollama.

[14] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi, Bertscore: Evaluating text generation with bert, in: International Conference on Learning Representations (ICLR), 2020. URL: https://openreview.net/forum?id=SkeHu1B6tS.

A. Prompts and tool scheme

This appendix provides system prompt templates in English and JSON schemas utilized during our evaluation. Figure 2 details the prompts used for the three operational modes of our system, while Figure 3 outlines the structured schema injected into the models for the "TOOL" execution mode.

[Cleaned text snippet up to 6000 characters from the source web page HTML]

"You are an information extraction assistant. You must respond fluidly with a short, direct answer. Do not write full sentences or give explanations.
You must use the `get_field` tool to read the structured data of the current entity. Base your answer solely on the result of the tool.

Question: What is the `[target_info]` of `[entity_label]`? The value must be short and directly comparable to Wikidata. The answer must be in `[language]`.
Extract the field `[target_info]`.
Wikidata field to extract: `[target_info]` Call `get_field` with this field, then give a short, direct answer.

Answer : "

Figure 2: Structure of the standardized prompt. Common elements (`no_rag` base) are shown in black, instructions for (`rag` mode) are in darkgreen and the instructions for (`tool_rag` mode) are highlighted in blue.

```
TOOL_SCHEMA = [
   {"type": "function",
      "function": {
         "name": "get_field",
         "description": "Returns the structured value of the
requested field for the current entity.",
         "parameters": {
            "type": "object",
            "properties": {
               "field_name": {
                  "type": "string",
                  "enum": TARGET_CHOICES,
                  "description": "Name of the field to read.",
               }
            },
            "required": ["field_name"], }, }, } ]
```

Figure 3: Structural JSON declaration schema for function calling in `tool_rag` mode.

B. Extended Experimental Results

Table 5 in this appendix presents the evaluation scores that back the visual trends summarized in Figure 2 of the main text as well as the gains of the RAG and TOOL modes. Mean BERTScore F_1 metrics are detailed comprehensively across all four target properties (Date of Birth, Place of Birth, Occupation, and Sex or Gender) and are further stratified by the model and the operational mode for every configuration. Table 6 summarizes the parameters of the configurations that attained the best BERTScore F_1 result in our experiment.

Table 5
Gain of RAG and TOOL RAG modes in mean BERTScore F_1 by property

Property	no rag	rag	rag gain	tool rag	tool rag gain	Mean by property
date_of_birth	0.791	0.738	-0.053	0.804	0.013	0.777
place_of_birth	0.720	0.837	0.117	0.763	0.043	0.773
occupation	0.669	0.721	0.052	0.714	0.045	0.701
sex_or_gender	0.788	0.828	0.04	0.824	0.036	**0.813**
Mean	0.742	**0.781**	**0.039**	0.776	0.035	

Table 6
Experimental Configurations Where the Top-Performing Model (Qwen) Achieved a Perfect BERTScore F_1 of 1.0

Target Property	Operational Mode	Prompt/Data Language	Citizenship Sample(s)
date_of_birth	tool_rag	Chinese	Eng, Fr, Ru, Zh
place_of_birth	tool_rag	Chinese	Fr
sex_or_gender	rag & tool_rag	Chinese	Eng, Fr, Ru, Zh