Mind the Language Gap in Digital Humanities: LLM-Aided Translation of SKOS Thesauri

Felix Kraus[1], Nicolas Blumenröhr[1], Danah Tonne[1], Achim Streit[1]

[1] Scientific Computing Center, Karlsruhe Institute of Technology, Karlsruhe, Germany

Abstract

We introduce WOKIE, an open-source, modular, and ready-to-use pipeline for the automated translation of SKOS thesauri. This work addresses a critical need in the Digital Humanities (DH), where language diversity can limit access, reuse, and semantic interoperability of knowledge resources. WOKIE combines external translation services with targeted refinement using Large Language Models (LLMs), balancing translation quality, scalability, and cost. Designed to run on everyday hardware and be easily extended, the application requires no prior expertise in machine translation or LLMs. We evaluate WOKIE across several DH thesauri in 15 languages with different parameters, translation services and LLMs, systematically analysing translation quality, performance, and ontology matching improvements. Our results show that WOKIE is suitable to enhance the accessibility, reuse, and cross-lingual interoperability of thesauri by hurdle-free automated translation and improved ontology matching performance, supporting more inclusive and multilingual research infrastructures.

Keywords: Translation System, Large Language Models, SKOS, Thesaurus, Multilingual

1 Introduction and Motivation

The organization and structuring of knowledge relies on controlled vocabularies, thesauri[1], and ontologies. These resources support the use of software for querying linked data, annotating datasets, and even forming core research objectives [11]. Within the Digital Humanities (DH), research communication and (metadata) publication frequently occur in English. This creates barriers for non-native English speakers and can limit or exclude them from participation and access to knowledge [30]. As a result, the culture, language, and history of these communities are under- or misrepresented, leading to a loss of cultural diversity and richness [30]. An important element to lower the barrier is to include additional languages in thesauri, particularly those relevant to the regions or communities connected to the research objects. Multilingual thesauri, often used for metadata, enable *collective benefit*, especially for those communities directly connected to the research, aligning with the *C* of the CARE principles[2].

As efforts by different communities grow to create multilingual thesauri, content overlaps are inevitable [8]. Additionally, different platforms often require slightly different terminologies, further contributing to knowledge fragmentation [23]. Ontology Matching (OM) addresses this heterogeneity by aligning equivalent terms or rather concepts[3]. However, multilingual thesauri present specific challenges for existing OM systems, particularly in the DH, where non-English content, historical languages, and various scripts are prevalent. Due to the predominance of English across most domains, current OM systems perform poorly on non-English or multilingual thesauri using

[1] A thesaurus arranges terms based on synonymy, hierarchical relationships and other properties. In contrast to ontologies, which offer comprehensive structures for semantic reasoning, thesauri primarily standardize terms.

[2] `https://www.gida-global.org/care`

[3] Although the used data model (SKOS) only defines concept as "an idea or notion; a unit of thought" [22], we use "term" instead within this work to avoid ambiguity.

the widespread Simple Knowledge Organization System (SKOS) model. The addition of English labels before matching is promising, which enables better performance of OM systems.

Achieving the translation step manually, whether for enabling multilingual access or improving OM, is not scalable. The use of classical statistical or neural Machine Translation (MT) methods for automated translation demands extensive bilingual corpora for supervised learning or monolingual data for unsupervised learning. Both approaches face significant limitations in DH contexts, where domain-specific bilingual corpora and training data are scarce [28], and unsupervised methods yield poor results in cases of domain mismatch and low-resource languages [15].

Given these limitations, Large Language Models (LLMs) have emerged as a promising method for translation [14]. Generic models do not require training by the individual researcher, but have the downside of longer processing times, possible hallucinations, and non-deterministic results. To counterfeit these issues while ensuring high-quality translations, we propose combining multiple external translation services with LLM-based refinement. Currently, no dedicated pipeline exists that integrates these two approaches specifically for translating SKOS thesauri.

To fill this gap, we introduce WOKIE (**W**ell-translated **O**ptions for **K**nowledge Management in **I**nternational **E**nvironments) which balances throughput, cost, and quality. WOKIE selectively applies LLM refinement when translations of multiple external services disagree. It uses configurable thresholds to only use LLMs when necessary, also reducing potential hallucinations by comparing LLM outputs directly to existing translation candidates. WOKIE is written in python, runs on everyday hardware, and can use solely free translation services. Translations generated by WOKIE are immediately integrated in the thesaurus, making the pipeline accessible to DH researchers without specialized infrastructure or extensive technical expertise.

The design of WOKIE is modular and easy-to-use: it applies a user-selected combination of eight primary translation services like Google translator, Lingvanex and ModernMT. Optionally, additional services can be implemented by using a common interface. The utilized LLM is chosen by the user out of 27 implemented models. Moreover, new models can be added easily, for example to use free or low-cost models provided by research institutions. This enables to tailor WOKIE to specific use cases.

In this work, we present the following main contributions, including an evaluation structured around five research questions (RQs):

- **WOKIE**, an open-source[4], modular and ready-to-use pipeline for automatic translation of SKOS thesauri, designed to run on everyday hardware and to be easily extendable.
- **RQ 1:** Which external translation services are most suitable as primary translators for the pipeline?
- **RQ 2:** How does LLM-based refinement impact the translation quality of SKOS thesauri?
- **RQ 3:** Which configuration parameters yield the best translation quality?
- **RQ 4:** Which LLMs are giving best results when used in the translation pipeline?
- **RQ 5:** What is the impact of pre-translation on ontology matching results?

2 Related Work

While no translation system specifically targets SKOS thesauri, several methods exist for translating ontologies and knowledge bases. Besides manual translation, different types of MT, including neural approaches, have been developed. Since they might be adoptable to the translation of DH SKOS thesauri, we review them below.

[4] `https://github.com/FelixFrizzy/WOKIE`

Early work on ontology label translation combined web services with contextual information from a database of multilingual ontologies [7]. It uses a semi-automatic approach presenting the user ambiguous cases, which is not scalable. More recently, a simpler approach without contextual information was implemented as a plugin for the ontology editor Protégé[5] [10], where disambiguation is not possible.

Another approach for translation is rule-based MT, relying on lexicons and rules defined by linguists [29]. In contrast, statistical MT and example-based MT methods learn from existing translations collected in bilingual corpora that include the domain of the translation task [25]. Statistical MT has been further enhanced by adding contextual information from lexicons [21] or parallel corpora [1], which showed slight improvements over standard statistical MT. These approaches require well-prepared datasets built by linguists or MT experts. In our case, such resources are scarce, and we aim for a method that can be applied by DH domain experts without extensive data preparation or detailed knowledge of machine translation.

Adaptive neural networks have also been proposed for knowledge base translation [9]. They represented subject and object triples as word embeddings and mapped them into a shared vector space using an embedding learning algorithm. Translation candidates obtained by an external service are ranked using these embeddings, making results highly dependent on external service accuracy. Furthermore, Neural MT requires pre-aligned training data for accurate results. Combining text- and triple-based models has shown better performance compared to the latter alone [24]. For the medical and financial domain, it has been shown that Neural MT models can be better fine-tuned for specific domains, leading to improved results compared to statistical MT [2].

To apply Neural MT to the DH domain where training data are scarce, architectural changes or adaptions are needed to achieve the same level of accuracy than in domains with more available training data [28]. For instance, intermediate fine-tuning steps have improved Neural MT results for French-Dutch translation in Fine Arts [3]. Unsupervised Neural MT avoids the need for parallel corpora, but the performance is affected whenever source and target monolingual data show linguistic differences and domain mismatch or low-resource languages are involved [15].

More recently, LLMs such as GPT-3 have been used for zero-shot translation. Due to their training on massive multilingual datasets [5], these models require no additional fine-tuning and are therefore accessible even for non experts in MT. GPT-4.5 for instance has demonstrated competitive results, outperforming some specialized neural MT systems, even leading German-English and English-German translation tasks [20]. Similarly, ChatGPT achieves translation quality comparable to commercial services, even for language pairs with substantial typological and syntactic divergence, such as Chinese and Romanian [14]. Their accessibility, broad language coverage and competitive results make them particularly relevant for our scenario. Therefore, they are examined in more detail in this work.

There exist also fine-tuned models for text-to-text translation, such as mT5 [32] and mBART [19]. However, they typically translate entire text inputs at once, making them unsuitable for translating individual terms enriched with context for disambiguation.

3 Translation Pipeline

3.1 Design Choices

WOKIE is a lightweight, modular pipeline designed to run efficiently on everyday hardware. The terms are translated individually, first with a user-selected set of external translation services, the primary services, which are generally fast and cost-efficient, or free. When necessary, an interchangeable LLM is employed for further refinement. It provides context-sensitive translation, but comes with higher latency, token-based costs, and usage limits. We utilize zero-shot prompts for

[5] `https://protege.stanford.edu/`

the LLM to keep WOKIE domain-agnostic and avoid complex fine-tuning, especially given the challenges in DH.

All implemented translators follow a common base class, which enables the easy integration of new translation services or LLMs. The supported languages depend solely on the chosen services, e.g. Google Translate supports 244 different languages, including Latin. We focus on the translation of `skos:prefLabel`, but other properties can be selected easily. The confidence threshold triggering LLM refinement is configurable: a higher threshold results in more frequent LLM usage, which increases execution time and potentially costs. The resulting translations are serialized back into an enriched thesaurus, enabling seamless use as a preprocessing step for any thesaurus creation, management, or alignment software.

3.2 Pipeline Overview

WOKIE processes a SKOS thesaurus using three main components: primary translators, confidence calculators and LLM-based refinement. A high-level overview of this pipeline is depicted in Figure 1. For each term and selected property (default is `skos:prefLabel`), WOKIE first collects primary translation candidates from the primary translation services. It then calculates a simple frequency-based confidence score. If the score meets or exceeds a user-defined threshold, the most frequent candidate is accepted directly. Otherwise, the LLM is used to get a context-sensitive secondary translation. If this matches a primary candidate, it is chosen as the final translation. If not, the LLM selects the best option of all primary and secondary candidates. The final translation is added back into the SKOS graph as a new literal in the target language.

3.2.1 Primary Translations

Users can select one or more primary translation services and prioritize their order. Each language label of a term is translated independently. Multiple services can be used to meet the minimum number of translations specified by the user. For monolingual thesauri, the maximum number of obtainable candidates equals the number of services that support the source-target language pair. The pipeline ensures translation of all labels, even if the minimum required translations are fewer. This ensures that as much information as possible is used for the translations.

3.2.2 Frequency-Based Confidence

A simple frequency method selects the most common translation based on exact string matches. Confidence is calculated as the number of occurrences of this translation divided by the total number of candidates. If the confidence is greater than or equal to the threshold, the translation is accepted immediately, bypassing the LLM refinement to save resources. Ambiguous cases, however, trigger further refinement.

3.2.3 LLM-Based Translation

Following the conclusions of §2, we make use of LLMs to support disambiguation. When primary confidence is too low, a prompt is constructed that includes the term description or definition, the general vocabulary description or a user selected context, depending on availability, for example:

> *Instructions*: You are a machine translation system that translates a term from any language to English.
> To determine the correct context, use the provided additional details. Return only the translated term and nothing else.
> *Input*: Term to translate: marginal gloss
> Description of the term that should be translated:
> A marginal gloss is a brief note in the margin explaining text.

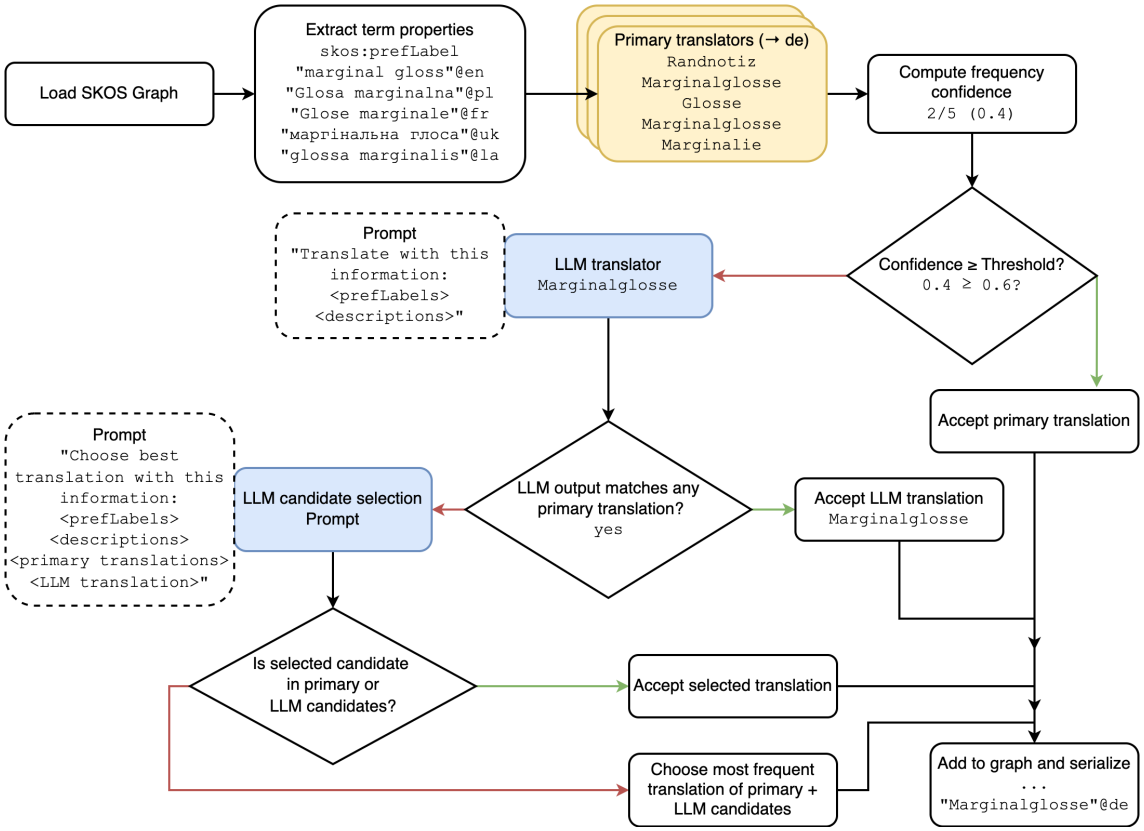

Figure 1: Simplified flowchart diagram of WOKIE's translation steps using "marginal gloss" as an example. The involvement of external translation services is marked in yellow, the use of LLMs blue. Positive decisions are on the right-hand side of the diamond (green arrow), negative decisions on the left (red arrow).

While some models are verbose, the translation is still extracted if possible. To make it resilient to hallucinated or inadequate translations, a final translation is only accepted if it matches any of the primary candidates.

3.2.4 LLM-Based Candidate Selection

If the LLM's translation is not in the primary translations, a prompt is used to choose the most accurate translation among all candidates. Parts about output format are spared out in the example:

> *Instructions*: You are a professional translation review system that assesses the quality of translations of a single term given in different source languages. The translations are already given by a translation system. Give me the best fitting translation out of the given list (...)
> Criteria for high accuracy are:
>
> - The best fitting translation is already found in the already given possible translations
> - In the current context, there is no possible translation that has a different meaning.
>
> Only give me the best fitting translation (...)
> *Input*: Choose the best fitting translation to German. (***first part identical to translation prompt***)
> The possible translations to German coming from translation systems are:
> Randnotiz, Marginalglosse, Glosse, Marginalie
> (...)

The following steps are similar to the LLM translation step. If parsing fails or the selection is invalid, a simple frequency-based calculation is used as a fallback option.

4 Evaluation Methodology

The primary goal of the evaluation is to compare various external translation services and LLMs and assess their suitability for translating DH thesauri within our pipeline. This goal is structured around the research questions posed at the end of §1.

4.1 Data Preparation and Translator Selection

As a first step, we identified and implemented 28 external translation services that provide an API, using wrapper libraries whenever possible. Due to issues such as low request limits, slow and unreliable responses, and other errors, eight services remained suitable for the evaluation. For selecting multilingual LLMs, we followed a similar process and also included models of a benchmark leaderboard [33] if possible. A comprehensive overview can be found in our GitHub repository[6] or in Appendix A.

For the accuracy comparison, we selected different multilingual DH thesauri, partly based on an existing multilingual DH ontology matching benchmark [16], see Table 1. We selected them because they are suitable for evaluating the effect of translation on ontology matching, and their size allows for repeated runs with different configurations and manual review. These thesauri use fifteen different languages across four scripts: Latin, Cyrillic, Greek, and Arabic. For each test case, we removed all properties in one language from a thesaurus and used WOKIE to translate them back. Then we compared the back-translated terms to the originals. When choosing which languages to remove, we excluded the language in which the thesaurus was originally developed. We determined this by examining early versions, related projects, or contacting developers. This avoided back-translating MT-generated label, preventing bias.

[6] `https://github.com/FelixFrizzy/WOKIE/blob/main/supported-services.md`

Resource	Version / Date	#terms[7]	language (ISO 639)
CodiKOS	- / -	∼170	de, en
DEFC Thesaurus	-	∼800	de, en, la
DYAS	3.1 / 2020-10-21	∼30	de, el, en, fr, la
iDAI.world Thesaurus	1.2 / 2022-02-10	∼270	de, en, es, fr, it
Iron-Age-Danube thesaurus	1 / 2018-11-07	∼290	de, en, hr, hu, sl
OeAI Thesaurus - Cultural Time Periods	1.0.0 / 2022-11-23	∼400	de, en
TaDiRAH	2.0.1 / 2021-07-22	∼170	de, en, es, fr, it, pt, sr
UNESCO	- / 2024-06-03	∼490	ar, en, fr, es, ru

Table 1: Thesauri used for the dataset.

4.2 Accuracy Measures

We used four different similarity measures to compare the original term with the back-translation, each normalized between 0 and 1, with the latter meaning identical strings. These measures provide relative comparisons rather than absolute similarity which is sufficient for our comparative analysis.

Exact Match A boolean test of string equality ignoring the case was used to find exact matches. This represents a conservative estimate of translation accuracy.

Levenshtein Similarity The Levenshtein or edit distance [18] counts the minimum number of character operations (removal, insertion, substitution) required to transform one string into another.

Jaro-Winkler Similarity The Jaro similarity [13] considers the number of common characters and transpositions between two strings, useful for measuring lexical similarity. The Winkler modification [31] improves the quality [6] by increasing the Jaro similarity when initial characters (up to four) match.

Cosine Similarity We chose the pre-trained BPEmb model [12], representing each word as an average of its subword vectors. Cosine similarity measures the semantic similarity based on the angle between the vectors. The smaller this angle, the higher the semantic similarity. BPEmb is best suited for our case because, for one part, using subword vectors prevents out-of-vocabulary errors that occur frequently on DH technical terms using word-based vectors. For the other, it is available in 275 languages, including Latin. Additionally, it has an over 540-times lower memory footprint than the comparable multilingual model fastText [4, 12]. This makes it ideal for running on everyday hardware, allowing users to perform tailored comparisons. LLM-based embeddings such as Gemini Embedding [17] were excluded to maintain independence from the evaluated LLMs.

4.3 Test Procedure

For each evaluation task, we selected thesauri samples across different languages and systematically varied one of the following parameters, keeping others constant:

- primary translation service,
- LLM,
- prompt composition,
- LLM temperature,

[7] Since we only used multilingual concepts, this number might be smaller than the unaltered thesaurus.

- confidence threshold, and
- minimum number of translations.

To ensure stability and reproducibility, some pipeline configurations were executed multiple times. We observed only negligible variations that have no impact on the results or their interpretation.

5 Evaluation Results

All scripts and thesauri used for the evaluation, their translations under various settings, computed distance measures, and related metadata are openly available as Zenodo record[8]. An example of translated labels can be found in Appendix A. The basic requirements for the evaluation of translation services and LLMs are a stable API, adequate request limits and reasonable output.

5.1 Comparison of Primary Translation Services

To address the first research question, we used 14 test cases consisting of thesauri that included only languages supported by all primary translation services. Figure 2 shows the macro average over all test cases (left) and micro averaged execution time per translation (right). Looking at string similarity, PONS, Argos, and Yandex consistently underperformed compared to other services. Similar is true for Levenshtein- and cosine similarity, although Yandex achieved better results for Jaro-Winkler similarity. For Latin translations, Google Translate performed best, with Lingvanex reaching between 75 and 93%, and ModernMT between 25 and 76% of Google's performance.

Comparisons across languages are challenging because the domain influences the results. However, looking at a single thesaurus only, comparisons are possible. For multilingual TaDiRAH, German and Serbian were the most difficult to translate accurately. For Latin within the DEFC Thesaurus, the string matches dropped to 2.38%, although cosine similarity was still at 0.25 which shows that translating Latin is possible, although it is likely that there is a bias introduced because Latin was most probably not the language in which it was developed.

Regarding execution times, the slowest, PONS, was eight times slower than Yandex with $0.22\,\mathrm{s}$ per translation. The execution times were generally consistent across languages for single services.

Due to occasional request limits encountered with ModernMT and Microsoft Translator, and to manage the usage of Google's paid service efficiently, we recommend the following prioritization for optimal translation outcomes: Lingvanex, Google Translate, ModernMT, Microsoft Translator, Yandex, Argos, Reverso, PONS.

5.2 Prompt Engineering

During our prompt experiments, several LLMs repeatedly failed to provide valid translations, instead generating irrelevant content such as repeating our question, definitions, or programming code. These models, of which most were free or older models like Llama or Mistral / Mixtral, were disregarded for further experiments. For models offering separate instruction and input prompts, we discovered that repeating instructions in the input leads to a more streamlined output.

We also evaluated three prompt strategies: individual, batch, and hierarchy. The individual approach translates all multilingual labels of one term individually, resulting in multiple translation candidates per term. The batch approach combines the information coming from all term labels into one prompt, producing a single translation candidate per term. The hierarchy approach expands the individual method by additionally including all broader terms up to the root term in the prompt.

[8] `https://doi.org/10.5281/zenodo.15494760`

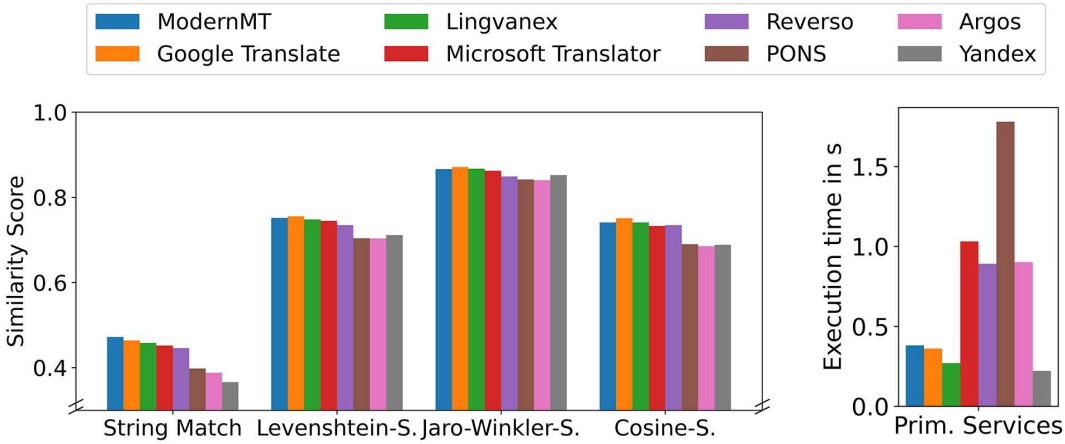

Figure 2: Comparison of the performance (left) and execution time (right) of different primary services, averaged over test cases.

None of the prompt strategies outperformed the others. The hierarchical method increased token usage without benefits because none of the tested representations of hierarchy relations could be correctly processed by the LLMs and quality did therefore not improve. The batch prompts provided only one translation per term, leading to less resilience to errors. We therefore chose the individual prompt approach for its simplicity and robustness.

5.3 Impact of Refinement

To assess the impact of LLM refinement (RQ2), we translated six thesauri using different pipeline configurations. Preliminary tests indicated that Gemini 2.0 Flash consistently ranked among the best-performing LLMs. We selected it as a representative model to ensure a fair comparison across the following pipeline configurations:

- single primary translator only,
- all recommended primary translators (see §5.1),
- single LLM only, no primary translators, and
- combination of all primary translators with LLM refinement.

Results in Figure 3 demonstrate that relying solely on one of the best-performing primary translators, Lingvanex, yields lower accuracy than other approaches. For configurations using all primary translators, we set the minimum number of required translations to five. This means that the pipeline stops querying once five translations are collected (or all labels are translated at least once), which does not always require all services to be called. In monolingual thesauri such as DEFC_de, this often leads to querying five different services, since only one source label is available per term. In contrast, multilingual thesauri like TaDiRAH_pt already contain labels in different languages. In such cases, a single service is sufficient to meet the minimum, as it can translate all labels across languages. As a result, the difference in translation quality between using one and all primary services is much smaller for these multilingual cases.

Most importantly, the results strongly suggest that either solely LLM-based translation or combining all primary translators with LLM refinement provides the best overall results. Surprisingly, Gemini 2.0 Flash underperforms when only used for translation to Serbian, but used just for refinement of difficult cases works still fairly well. Given that LLMs typically have higher latency,

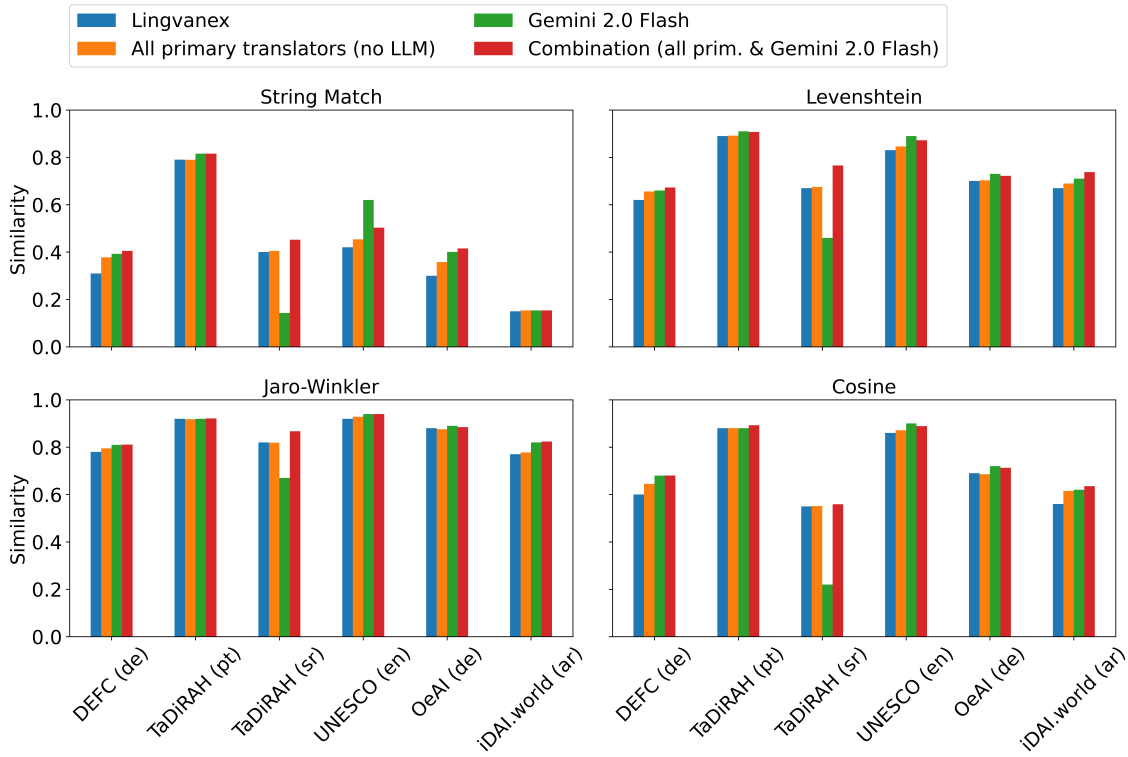

Figure 3: Similarity measures for the four test setups and multiple test cases. The removed and back-translated language is indicated in parentheses.

cost, and unclear training language coverage, the optimal configuration uses external translation services with targeted LLM refinement for challenging cases.

Across test cases, the four similarity measures tend to align in relative terms: when one measure indicates low or high similarity, the others usually follow the same trend. This consistency, though not exact in absolute values, supports the qualitative reliability of the findings.

5.4 Parameter Tuning

To determine the optimal parameter set (RQ3), we conducted a series of experiments. For each, we used a set of base settings: a threshold of 0.6, a minimum of five primary translations, a temperature of 0 and using the individual prompt strategy. We selected six LLMs from six different providers that had shown promising results in preliminary tests, based on translation quality, rate limits, and token pricing. We also reduced the number of test cases to three to allow a manageable amount of different runs.

Temperature The temperature setting influences the randomness and creativity of an LLM's output. Therefore, we tested the impact of temperature values 0 (close to deterministic), 0.5 and 1 (more variable) on the translation results. Across most models, we observed no notable performance differences. This is likely due to the short length of the output, which limits the model's opportunity for variation. Our findings align with the literature, where no significant impact on problem-solving tasks within this temperature range was reported [27].

Threshold The confidence threshold is closely tied to the number of primary translations, which is kept at five for these experiments. We chose to set the threshold to 0.4, 0.6 and 0.8, which

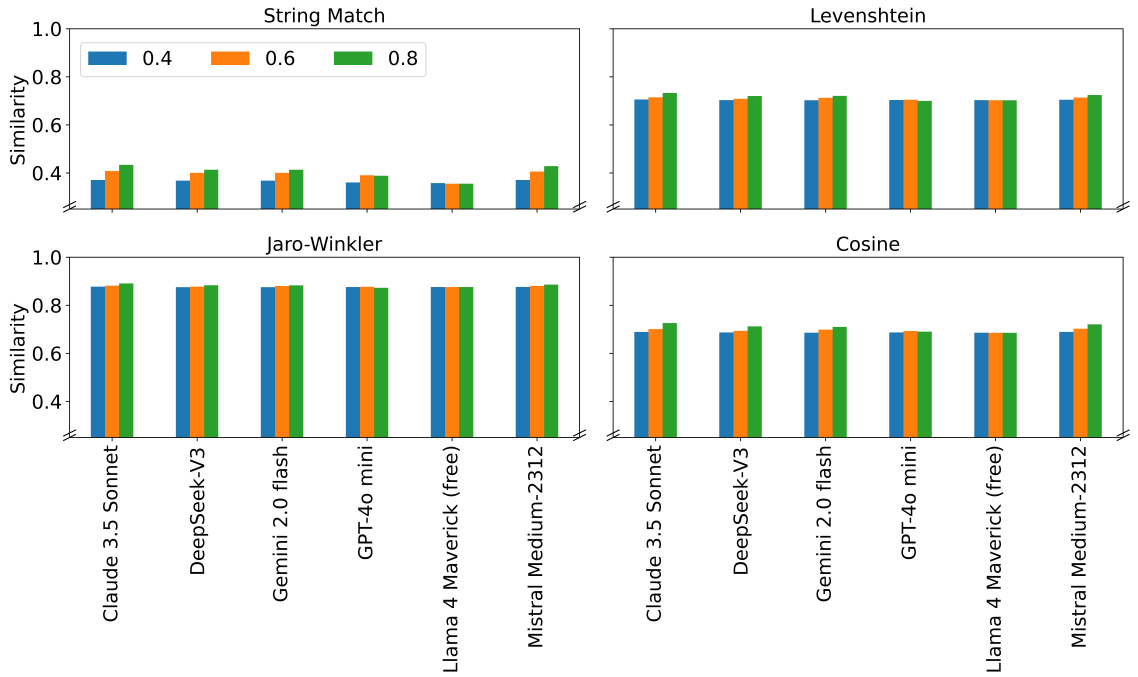

Figure 4: Similarity measures for different threshold values using OeAI Thesaurus back-translated to German.

corresponds to accepting a translation directly if at least two, three, or four out of five primary translation candidates agree. As shown in Figure 4, increasing the threshold slightly improved translation quality for some models. A similar conclusion can be drawn for tests with other thesauri.

The higher the threshold, the more often the LLM is called. The total running time, which is proportional to the number of LLM calls, rises only slightly when raising the threshold from 0.4 to 0.6. However, moving to 0.8 significantly raised execution time by roughly 50–100%, depending on the model. Given the minimal quality improvement beyond 0.6, this higher cost is not justified. We therefore selected 0.6 as the optimal threshold.

Minimum Number of Primary Translations As mentioned in §3.2.1, users can set the minimum number of primary translations, although the pipeline always ensures that at least all labels of a term are translated once. Because this parameter is linked to the confidence threshold, isolating its effect was not possible. We therefore only performed selected tests with values of three, five, and eight. These sometimes showed either a slight decrease in accuracy for values of three or eight, depending on the test case and language model. Therefore, we selected the minimum number of primary translations to five, which consistently showed good results. We conclude that at least three identical translation candidates are recommended for direct acceptance as final translation. This is the sweet spot between accuracy and resources, so the product of threshold and minimum translations should be no less than three.

5.5 LLM Comparison

Having shown the general benefit of LLM-based refinement, the fourth research question about the best fitting models becomes apparent. We used the optimal parameters and translation services identified earlier for this evaluation. Figure 5 presents the averaged results across several test cases.

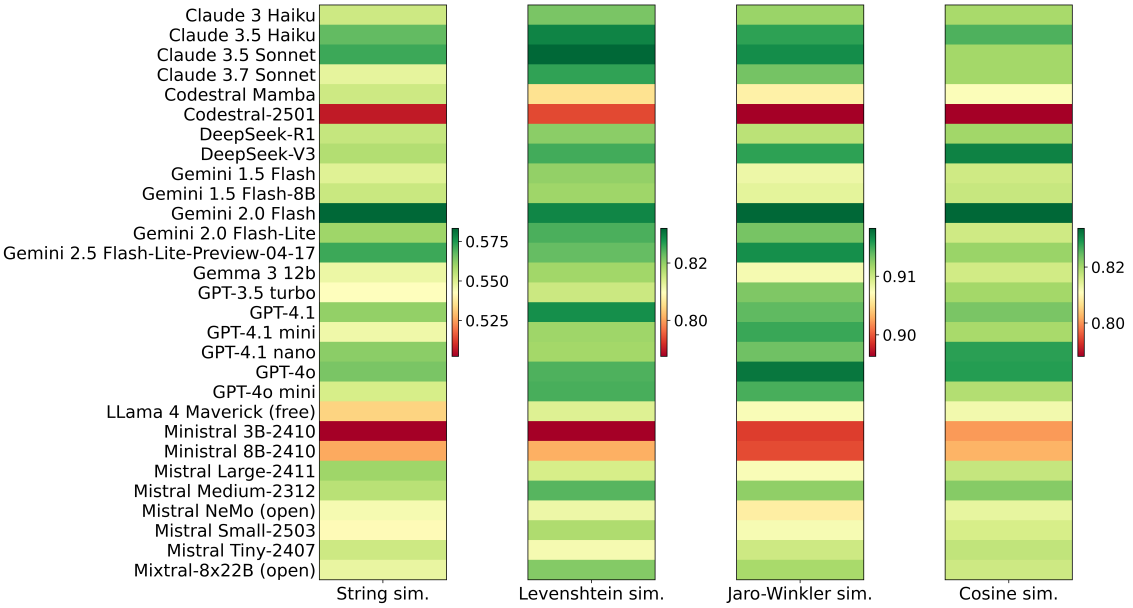

Figure 5: Similarity measures for different LLMs averaged over test cases. Green indicates the best values, transitioning over yellow to red for lower scores.

Additional details on pricing and availability can be found in the GitHub repository[9].

The similarity scores indicate a group of well-performing models: Gemini 2.0 Flash, DeepSeek-V3, GPT-4o, Claude 3.5 Haiku, Gemini 2.5 Flash-Lite-Preview-04-17, and Claude 3.5 Sonnet. Gemini 2.0 Flash stands out within this group for having a comparable low token cost, while either leading or being among the best regarding the similarity measures. Moreover, its API provides high request limits, which we never exceeded in our tests. In contrast, we frequently hit rate limits when using DeepSeek and Mistral models. Based on performance and practical considerations, we recommend Gemini 2.0 Flash. Among the openly available models, DeepSeek-V3 shows clearly the best results.

Using Gemini 2.0 Flash, we translated all 19 thesauri with a total of 6475 terms, many with multilingual labels, in under 3.3 hours. This strongly suggests that WOKIE is well suited for small to medium-sized thesauri, as commonly created and used in the DH. We also examined results for Latin separately, given its relevance in DH. All LLM-enhanced combinations outperformed the best single primary service in all similarity measures. The best ones (GPT-4.1 mini, Gemini 2.0 Flash) achieved up to 0.31 string similarity, compared to 0.04 without LLM. This highlights the clear benefit of refinement for Latin.

5.6 Ontology Matching

WOKIE functions as a preprocessing step, therefore existing OM systems can benefit from translations without modifications. To answer the last research question regarding the impact of English pre-translation on OM, we used a multilingual SKOS benchmark from archaeology [26]. All but one language were removed from the thesauri, then the matching was performed, and the alignment compared to ground truth alignments. We reproduced the benchmark using two existing matching systems and compared it to the results when pre-translation was performed with WOKIE. We used the F1 score (the harmonic mean of precision and recall) as our evaluation metric, as commonly done in OM benchmarking.

[9] `https://github.com/FelixFrizzy/WOKIE/blob/main/supported-services.md`

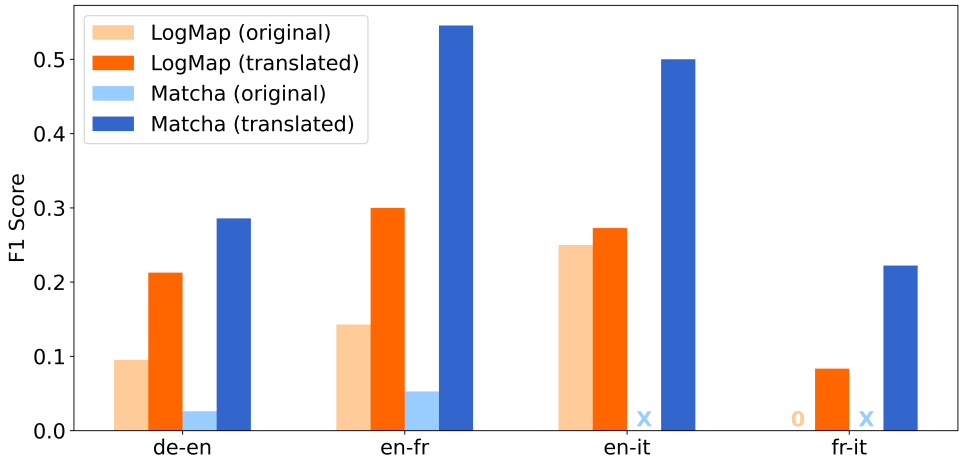

Figure 6: Comparison of the F1 scores from the original benchmark "(original)" with the results obtained after translating all thesauri to English first "(translated)". F1 scores of zero are denoted with a "0", errors on execution of the matching system with "X".

As shown in Figure 6, results improved for all language pairs after translation. The highest F1 score gain was observed using Matcha: an increase of 0.49 for English-French and a successful execution of previously failing English-Italian with an F1 score of 0.5. Even for the French-Italian pair, where both thesauri were translated, the F1 score for Matcha rises to 0.22. These findings demonstrate that introducing a preprocessing step for non-English thesauri can significantly improve matching results with minimal effort.

6 Discussion and Limitations

Despite our best efforts, testing all combinations of parameters, primary translation services, and LLMs was not feasible due to the large number of possible configurations and therefore exceeding rate limits. Additionally, it was not possible to include more languages, primarily due to the limited availability of high-quality multilingual thesauri. We showed that our pipeline is even applicable to the lower-resource language Serbian. Therefore, it is justified to assume that the pipeline also produces meaningful translations for untested languages, as long as they are supported by a subset of primary services and the selected LLM. Looking at domain specifics, we focused on those where domain-specific thesauri were available, and therefore do not claim to cover the wide range of domains belonging to or using elements of the DH.

When examining multilingual labels used as ground truth, it becomes clear that this is not always justified. For example, in a thesaurus about early farming cultures, there is a term describing pottery decorations which has the English label "red on red (a3γ)" and a German label "Rot on red (a3c)". Our pipeline correctly translated it to "rot auf rot (a3γ)", which is nevertheless penalized because of an incorrect reference label. As a result, some systems may show lower similarity measures. It is also important to note that whenever automatic translation systems are involved, we recommend to either review the translations by experts or explicitly flag them as machine-generated. Looking at the quality of SKOS thesauri in general, we located multiple issues when adhering to the data model. For instance, one thesaurus used URLs linking to Wikidata entities in the `skos:definition` field instead of the more appropriate `skos:related` or `skos:exactMatch` properties. Consequently, the pipeline performance is limited by the quality of the input data.

7 Conclusions and Outlook

In this work, we introduced WOKIE, a modular and ready-to-use translation pipeline designed to support the multilingual needs of the DH community. WOKIE provides scalable translations of SKOS thesauri without requiring prior knowledge of SKOS, MT or LLMs. We systematically evaluated combinations of services, LLMs and parameter settings to identify the optimal configuration of WOKIE for DH thesauri.

Our evaluation demonstrated that WOKIE enables accurate, automatic and scalable translation of DH thesauri. The integration of LLM-based refinement substantially improved translation quality, especially in ambiguous cases. We further showed that pre-translating thesauri into English prior matching significantly enhances the performance of ontology matching systems. In one benchmark, the F1-score rose from zero to 0.5 after translation.

Looking at the scalability, we showed that translating thesauri with up to 10,000 terms is feasible when using services and models with sufficient request limits. To extend this further, we plan to integrate load balancer capabilities to handle even larger datasets.

WOKIE lowers technical barriers for multilingual vocabularies, supporting inclusive metadata practices and enabling better representation of communities with lower levels of English proficiency. Moreover, we plan to evaluate and possibly integrate synonyms provided by translations services. Since objective, automated, quantitative metrics for the translation quality of descriptions are difficult, if not impossible, we plan manual reviews including domain experts on small thesauri.

Lastly, we aim to explore the impact on OM more broadly by investigating observed effects and testing additional matching systems with different datasets. To achieve this, we plan to include all suitable multilingual thesauri and matching systems participating in the recent campaigns of the Ontology Alignment Evaluation Initiative (OAEI). Overall, we believe that WOKIE contributes to more equitable and language-aware knowledge infrastructures, a key step toward inclusive and language-independent reuse of research data.

Acknowledgements

This research was funded by the research program “Engineering Digital Futures” of the Helmholtz Association of German Research Centers, and the Helmholtz Metadata Collaboration Platform (HMC).

Disclosure. Since the author’s mother tongue is not English, LLM assisted tools such as LanguageTool were used to improve punctuation, grammar, and spelling in full compliance with the Large Language Model Policy of the conference. In no way was any content created by these tools.

References

[1] Arcan, Mihael and Buitelaar, Paul. “Ontology Label Translation.” In: *Proceedings of the 2013 NAACL HLT Student Research Workshop*, ed. by Annie Louis et al. Atlanta, Georgia: Association for Computational Linguistics, June 2013, pp. 40–46.

[2] Arcan, Mihael, Torregrosa, Daniel, and Buitelaar, Paul. “Translating Terminological Expressions in Knowledge Bases with Neural Machine Translation.” July 2019. DOI: 10.48550/arXiv.1709.02184. arXiv: 1709.02184 [cs].

[3] Banar, Nikolay et al. “Transfer Learning for Digital Heritage Collections: Comparing Neural Machine Translation at the Subword-level and Character-level:” in: *Proceedings of the 12th International Conference on Agents and Artificial Intelligence*. Valletta, Malta: SCITEPRESS - Science and Technology Publications, 2020, pp. 522–529. ISBN: 978-989-758-395-7. DOI: 10.5220/0009167205220529.

[4] Bojanowski, Piotr et al. “Enriching Word Vectors with Subword Information.” In: *Transactions of the Association for Computational Linguistics 5* (2017), ed. by Lillian Lee, Mark Johnson, and Kristina Toutanova, pp. 135–146. DOI: 10.1162/tacl_a_00051.

[5] Brown, Tom B. et al. “Language Models Are Few-Shot Learners.” In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*. NIPS ’20. Red Hook, NY, USA: Curran Associates Inc., Dec. 2020, pp. 1877–1901. ISBN: 978-1-7138-2954-6.

[6] Christen, Peter. “A Comparison of Personal Name Matching: Techniques and Practical Issues.” In: *Sixth IEEE International Conference on Data Mining - Workshops (ICDMW’06)*. Hong Kong, China: IEEE, 2006, pp. 290–294. ISBN: 978-0-7695-2702-4. DOI: 10.1109/ICDMW.2006.2.

[7] Espinoza, Mauricio, Gómez-Pérez, Asunción, and Mena, Eduardo. “LabelTranslator - A Tool to Automatically Localize an Ontology: 5th European Semantic Web Conference, ESWC 2008.” In: *The Semantic Web*. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) (2008), pp. 792–796. ISSN: 3540682333. DOI: 10.1007/978-3-540-68234-9_60.

[8] Euzenat, Jérôme and Shvaiko, Pavel. *Ontology Matching*. second edition. Berlin, Heidelberg: Springer, 2013. ISBN: 978-3-642-38721-0. DOI: 10.1007/978-3-642-38721-0.

[9] Feng, Xiaocheng et al. “English-Chinese Knowledge Base Translation with Neural Network.” In: *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, ed. by Yuji Matsumoto and Rashmi Prasad. Osaka, Japan: The COLING 2016 Organizing Committee, Dec. 2016, pp. 2935–2944.

[10] Florrence, Merlin. “MLGrafViz: Multilingual Ontology Visualization Plug-in for Protégé.” In: *Computer Science and Information Technologies 2.*, no. 1 (Mar. 2021), pp. 43–48. ISSN: 2722-3221. DOI: 10.11591/csit.v2i1.p43-48.

[11] Haslhofer, Bernhard, Isaac, Antoine, and Simon, Rainer. “Knowledge Graphs in the Libraries and Digital Humanities Domain.” In: *Encyclopedia of Big Data Technologies*, ed. by Sherif Sakr and Albert Zomaya. Cham: Springer International Publishing, 2018, pp. 1–8. ISBN: 978-3-319-63962-8. DOI: 10.1007/978-3-319-63962-8_291-1.

[12] Heinzerling, Benjamin and Strube, Michael. “BPEmb: Tokenization-free Pre-trained Subword Embeddings in 275 Languages.” In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, ed. by Nicoletta Calzolari et al. Miyazaki, Japan: European Language Resources Association (ELRA), May 2018.

[13] Jaro, Matthew A. “Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida.” In: *Journal of the American Statistical Association 84.*, no. 406 (June 1989), pp. 414–420. ISSN: 0162-1459. DOI: 10.1080/01621459.1989.10478785.

[14] Jiao, Wenxiang et al. “Is ChatGPT A Good Translator? Yes With GPT-4 As The Engine.” Nov. 2023. DOI: 10.48550/arXiv.2301.08745. arXiv: 2301.08745 [cs].

[15] Kim, Yunsu, Graça, Miguel, and Ney, Hermann. “When and Why Is Unsupervised Neural Machine Translation Useless?” In: *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, ed. by André Martins et al. Lisboa, Portugal: European Association for Machine Translation, Nov. 2020, pp. 35–44.

[16] Kraus, Felix et al. “A Gold Standard Benchmark Dataset for Digital Humanities.” In: *Proceedings of the 19th International Workshop on Ontology Matching*, ed. by Ernesto Jiménez-Ruiz et al. Vol. 3897. CEUR Workshop Proceedings. Baltimore, MD, USA: CEUR, Nov. 2024, pp. 1–17. DOI: 10.5445/IR/1000178023.

[17] Lee, Jinhyuk et al. “Gemini Embedding: Generalizable Embeddings from Gemini.” Mar. 2025. DOI: 10.48550/arXiv.2503.07891. arXiv: 2503.07891 [cs].

[18] Levenshtein, Vladimir I. “Binary Codes Capable of Correcting Deletions, Insertions, and Reversals.” In: *Soviet Physics Doklady 10.*, no. 8 (1966), pp. 707–710.

[19] Liu, Yinhan et al. “Multilingual Denoising Pre-training for Neural Machine Translation.” In: *Transactions of the Association for Computational Linguistics 8* (Dec. 2020), pp. 726–742. ISSN: 2307-387X. DOI: 10.1162/tacl_a_00343.

[20] Manakhimova, Shushen et al. “Linguistically Motivated Evaluation of the 2023 State-of-the-art Machine Translation: Can ChatGPT Outperform NMT?” In: *Proceedings of the Eighth Conference on Machine Translation*. Singapore: Association for Computational Linguistics, 2023, pp. 224–245. DOI: 10.18653/v1/2023.wmt-1.23.

[21] Mauser, Arne, Hasan, Saša, and Ney, Hermann. “Extending Statistical Machine Translation with Discriminative and Trigger-Based Lexicon Models.” In: *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, ed. by Philipp Koehn and Rada Mihalcea. Singapore: Association for Computational Linguistics, Aug. 2009, pp. 210–218.

[22] Miles, Alistair and Bechhofer, Sean. *SKOS Simple Knowledge Organization System Reference*. W3C Recommendation. United States: World Wide Web Consortium, Aug. 2009.

[23] Morvillo, Alberto and Mecella, Massimo. “Integrating Multiple Knowledge Graphs in Digital Humanities.” In: *ST4DM 2024: Semantic Technologies for Data Management*. Twente, Italy, July 2024.

[24] Moussallem, Diego, Soru, Tommaso, and Ngonga Ngomo, Axel-Cyrille. “THOTH: Neural Translation and Enrichment of Knowledge Graphs.” In: *The Semantic Web – ISWC 2019: 18th International Semantic Web Conference, Auckland, New Zealand, October 26–30, 2019, Proceedings, Part I*. Berlin, Heidelberg: Springer-Verlag, Oct. 2019, pp. 505–522. ISBN: 978-3-030-30792-9. DOI: 10.1007/978-3-030-30793-6_29.

[25] Moussallem, Diego, Wauer, Matthias, and Ngomo, Axel-Cyrille Ngonga. “Machine Translation Using Semantic Web Technologies: A Survey.” In: *Journal of Web Semantics 51* (Aug. 2018), pp. 1–19. ISSN: 15708268. DOI: 10.1016/j.websem.2018.07.001. arXiv: 1711.09476 [cs].

[26] Pour, Mina Abd Nikooie et al. “Results of the Ontology Alignment Evaluation Initiative 2024.” In: *Proceedings of the 19th International Workshop on Ontology Matching (OM 2024), Baltimore, USA, November 11, 2024. Ed.: E. Jiménez-Ruiz, O. Hassanzadeh, C. Trojahn, S. Hertling, H. Li, P. Shvaiko, J. Euzenat*. Vol. 3897. CEUR Workshop Proceedings. Baltimore, MD, USA: CEUR, Nov. 2024, pp. 64–97. DOI: 10.5445/IR/1000179469.

[27] Renze, Matthew. “The Effect of Sampling Temperature on Problem Solving in Large Language Models.” In: *Findings of the Association for Computational Linguistics: EMNLP 2024*, ed. by Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen. Miami, Florida, USA: Association for Computational Linguistics, Nov. 2024, pp. 7346–7356. DOI: 10.18653/v1/2024.findings-emnlp.432.

[28] Suissa, Omri, Elmalech, Avshalom, and Zhitomirsky-Geffet, Maayan. “Text Analysis Using Deep Neural Networks in Digital Humanities and Information Science.” In: *Journal of the Association for Information Science and Technology 73*., no. 2 (2022), pp. 268–287. ISSN: 2330-1643. DOI: 10.1002/asi.24544.

[29] Thurmair, Gregor. “Comparing Rule-Based and Statistical MT Output.” In: *Proceedings of the LREC 2004 Workshop on The Amazing Utility of Parallel and Comparable Corpora*. Lissabon, 2004, pp. 5–9.

[30] Viola, Lorella. “Editorial: Data and Workflows for Multilingual Digital Humanities.” In: *Journal of Open Humanities Data 10* (June 2024), p. 37. ISSN: 2059-481X. DOI: 10.5334/johd.220.

[31] Winkler, William E. “String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage.” In: *Proceedings of the Survey Research Methods Section, ASA (1990)* (Jan. 1990), pp. 354–359.

[32] Xue, Linting et al. “mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer.” In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, 2021, pp. 483–498. DOI: 10.18653/v1/2021.naacl-main.41.

[33] Zhou, Yi et al. “Multilingual MMLU Benchmark Leaderboard.” 2024.

Acronyms

DH Digital Humanities

LLM Large Language Model

MT Machine Translation

OAEI Ontology Alignment Evaluation Initiative

OM Ontology Matching

RQ research question

SKOS Simple Knowledge Organization System

A Tables

Name	Implemented?	Free?	Comment
Argos	✔	✔	To be used locally with LibreTranslate (API of Argos)
Google Translate	✔	✖	Uses Cloud Translation API
Lingvanex	✔	✔	
	✔	✔	
Microsoft Translator	✔	✖	Quite low request limits
PONS	✔	✖	No latin and serbian in contrast to the PONS online dictionary
Reverso	✔	✔	Sometimes no API response
Yandex Translate	✔	✔	
Alibaba	✖		Very slow
Apertium	✖		Unexpected exception when using the API
BabelNet	✖		Very low request limit
Bing	✖		Very low request limit
Caiyun	✖		Very low requests per second
CloudTranslation	✖		Very low requests per second
DeepL	✖		Very low requests per second
elia	✖		Very slow
hujiang	✖		Very low requests per second
iTranslate	✖		Very low requests per second
languageWire	✖		Very low requests per second
Linguee	✖		Very low requests per second
Mymemory	✖		Very low request limit
OpenNMT	✖		Only for full texts
Papago	✖		Very slow, mainly for Korean
QcriTranslator	✖		Obligatory registration failed
qqTranSmart	✖		Very slow
Sogou	✖		Very slow
Tencent	✖		Identical to sogou
TranslateCom	✖		Very low request limit

Table 2: Comparison of Translation APIs

Costs as they were on 1st of May 2025 using the API provided by the manufacturer.

[10] The input tokens dominate the costs largely in WOKIE, which is why the total costs for a thesaurus are only calculated with the input costs.

Model Name	Implemented?	Input Costs (USD/1M tk)	Costs for about 100 terms in ¢[10]	Free to download?	Comment
claude-3-5-haiku	✔	0.80	2.4	✖	
claude-3-5-sonnet	✔	3	9	✖	
claude-3-7-sonnet	✔	3	9	✖	
claude-3-haiku	✔	0.25	0.75	✖	
codestral-latest	✔	0.3	0.9	✖	Pointing to Codestral-2501
deepseek-chat	✔	0.27	0.81	✔	Pointing to DeepSeek-V3
deepseek-reasoner	✔	0.14	0.42	✔	Pointing to DeepSeek-R1
gemini-1.5-flash	✔	0.075	0.225	✖	
gemini-1.5-flash-8b	✔	0.0375	0.1125	✖	
gemini-2.0-flash	✔	0.10	0.3	✖	
gemini-2.0-flash-lite	✔	0.075	0.225	✖	
gemini-2.5-flash-preview-04-17	✔	0.15	0.45	✖	
gemma3:12b	✔	0	0	✔	
gpt-3.5-turbo	✔	0.50	1.5	✖	
gpt-4.1-mini	✔	0.40	1.60	✖	
gpt-4.1-nano	✔	0.10	0.3	✖	
gpt-4.1	✔	Unknown	Unknown	✖	
gpt-4o	✔	2.50	7.5	✖	
gpt-4o-mini	✔	0.15	0.45	✖	
llama-4-maverick:free	✔	0	0	✔	
ministral-3b-latest	✔	0.04	0.12	✖	Pointing to Ministral 3B-2410
mistral-large-latest	✔	2	6	✖	Pointing to Mistral Large-2411
mistral-medium-latest	✔	0.4	1.2	✖	Pointing to Mistral Medium-2312
mistral-tiny-latest	✔	Unknown		✖	Pointing to Mistral Tiny-2407
mistral-small-latest	✔	0.15	0.45	✔	Pointing to Mistral Small-2503
open-mistral-nemo	✔	0.15	0.45	✔	
open-mixtral-8x22b	✔	2	6	✔	
codestral-mamba-latest	✖	Unknown	Unknown	✔	Poor performance; Pointing to Codestral Mamba
llama-4-scout:free	✖	0	0	✔	Cannot be reliably constrained to adhere to the expected output format
llama3.2 (ollama)	✖	0	0	✔	Returned output is irrelevant and lacks meaningful content
ministral-8b-latest	✖	0.1	0.3	✖	Poor performance; Pointing to Ministral 8B-2410
open-codestral-mamba	✖			✔	Cannot be reliably constrained to adhere to the expected output format, also overly verbose
open-mistral-7b	✖			✔	Cannot be reliably constrained to adhere to the expected output format, also overly verbose
open-mixtral-8x7b	✖	0.7	0.21	✔	Cannot be reliably constrained to adhere to the expected output format, also overly verbose

Table 3: Comparison of LLMs

Original label (en)	Original label (de)	Translated label by WOKIE (de)
Analyzing	Analyse	Analyse
Annotating	Annotieren	Anmerkung
Archiving	Archivieren	Archivierung
Capturing	Erfassen	Erfassung
Collaborating	Kollaboration	Zusammenarbeit
Commenting	Kommentieren	Kommentar
Communicating	Kommunizieren	Kommunikation
Content Analysis	Inhaltsanalyse	Inhaltsanalyse
Contextualizing	Kontextualisieren	Kontextualisierung
Converting	Konvertieren	Konvertierung
Creating	Erzeugen	Schöpfung
Crowdsourcing	Crowdsourcing	Crowdsourcing
Data Cleansing	Bereinigen	Datenbereinigung
Data Recognition	Datenerkennung	Datenerkennung
Designing	Design	Design
Discovering	Entdecken	Entdeckung
Disseminating	Dissemination	Verbreitung
Editing	Bearbeiten	Bearbeitung
Enriching	Anreichern	Anreicherung
Gathering	Sammeln	Sammlung
Identifying	Identifizieren	Identifizierung
Imaging	Bilderfassung	Bildgebung
Interpreting	Interpretation	Interpretation
Modeling	Modellieren	Modellierung
Network Analysis	Netzwerkanalyse	Netzwerkanalyse
Organizing	Organisieren	Organisieren
Preserving	Konservierung	Erhaltung
Programming	Programmieren	Programmierung
Publishing	Veröffentlichen	Veröffentlichung
Recording	Aufzeichnen	Aufnahme
Relational Analysis	Analyse von Relationen	Relationale Analyse
Sharing	Teilen	Teilen
Spatial Analysis	Räumliche Analyse	Raumanalyse
Storing	Speicherung	Speicherung
Structural Analysis	Strukturanalyse	Strukturanalyse
Stylistic Analysis	Stilistische Analyse	Stilanalyse
Theorizing	Theoriebildung	Theoretisierung
Transcribing	Transkription	Transkription
Translating	Übersetzen	Übersetzung
Visual Analysis	Visualisierung	Visualisierung
Web Development	Webentwicklung	Webentwicklung
Writing	Schreiben	Schreiben

Table 4: Original English, original German and translated German labels of TaDiRAH. The translation was obtained by WOKIE without knowledge of the original German label.