Comparing Human and AI-Assisted Evaluation of Interpreting Performance: A Diagnostic Study Using ChatGPT-4o

Tomasz Korybski[*], University of Warsaw
Karolina Broś[**], University of Warsaw
Małgorzata Szupica-Pyrzanowska[***], University of Warsaw

ABSTRACT

Growing data volumes and the rise of AI-assisted interpreting solutions increasingly challenge the assessment of interpreting quality. This study investigates whether large language models (LLMs) can support error-based interpreting performance assessment, which has traditionally relied on human-driven approaches. We used transcripts from 40 English-to-Polish interpreting outputs (Broś et al., 2025), evaluated using the NTR model (Romero-Fresco & Pöchhacker, 2017). Two trained human evaluators conducted detailed analyses, which were then compared against ChatGPT-4o outputs generated through different prompting strategies. Initial results showed an overlap (recall) of up to 85% in omission detection between human and AI assessments when a full prompting sequence was applied and the source and target texts were provided in the chat's interface rather than in uploaded text files. However, bulk and iterative analyses revealed significant limitations, including prompt drift and reduced accuracy over successive tasks, with the lowest overlaps below 10%. Despite these drawbacks, the AI model occasionally identified errors overlooked by humans and demonstrated potential for expediting error annotation. The study highlights that, while LLMs provide partial automation and enhance consistency, comprehensive human oversight remains indispensable. Ultimately, integrating AI with human expertise can become a promising hybrid approach to interpreting quality evaluation, balancing efficiency with nuanced judgment.

KEYWORDS

Interpreting quality, large language models, ChatGPT-4o, NTR model, AI-assisted evaluation, interpreting performance.

1. Introduction

The evaluation of interpreting quality remains a vital aspect of interpreting research and practice, particularly considering the exponential growth in interpreting data and the advent of new interpreting technologies. Computer-Aided Interpreting (CAI) tools and Machine Interpreting (MI, commonly branded by service providers as 'AI Interpreting') change the way the service is delivered (Prandi, 2025), contributing to the pertinence of questions on different aspects of output quality in interpreting, and on the need for streamlined evaluation approaches. The incorporation of technology into interpreting practice, including remote and machine-assisted interpreting, necessitates new frameworks for interpreter performance assessment that reflect contemporary working conditions and technological affordances (Davitti et al., 2025). At the same time, the advent of powerful Large Language Models (LLMs) is opening opportunities for more efficient and faster analyses of large bilingual datasets. The speed factor is particularly relevant: as technology companies release further iterations of their automated interpreting solutions, there is a pressing need for speedy and precise scrutiny of outputs generated by such solutions, as well as their benchmarking against human performance. Against this backdrop, the present paper examines

[*] ORCID https://orcid.org/0000-0003-2353-0816, t.korybski@uw.edu.pl
[**] ORCID https://orcid.org/0000-0001-6701-9257, k.bros@uw.edu.pl
[***] ORCID https://orcid.org/0000-0003-4067-4483, m.szupica-pyrz@uw.edu.pl

https://doi.org/10.26034/cm.jostrans.2026.8425

whether LLMs can perform source-target comparisons in interpreting at a level comparable to human evaluators. We strive to present a balanced view of the advantages and constraints of both human-driven and automated assessments built around the well-established concept of error-based accuracy evaluation – one of the key aspects of interpreting quality.

2. Quality Evaluation in Interpreting Contexts

Notably, approaches successfully applied in the evaluation of translation do not necessarily work for interpreting data. Although the broad categories of errors (e.g., Barik, 1971) can apply to both translation and interpreting outputs, it is important to consider the differences arising from the distinct settings and contexts of translation and interpreting.

Automated metrics developed for machine translation (MT), such as BLEU (Papineni et al., 2002) and METEOR (Banerjee & Lavie, 2005), may encounter problems when applied to interpreting data, given the fundamental differences inherent in translation and interpreting. Specifically, automated MT evaluation metrics operationalise quality in terms of reference-based surface or semantic similarity, whereas interpreting is constrained by real-time processing and involves legitimate divergence strategies. These include summarisation, paraphrase, segmentation, omission, and sometimes even expansion and/or explicitation. All these strategies preserve communicative adequacy while reducing reference-overlap scores.

In this regard, some studies have shown that off-the-shelf MT metrics correlate with human ratings under specific conditions such as consecutive or controlled settings (Lu & Han, 2023; Macháček et al., 2023). Lu & Han (2023) report positive results but limit their claims to the consecutive mode and call for multiple replications prior to a large-scale study based on hundreds of recordings across different language pairs and settings. Macháček and colleagues (2023) also report positive results but base their analysis on a very small reference dataset – one interpreter's performance. It is worth emphasising that, while these studies are valuable and show promise, they do not fully address the specific challenges posed by simultaneous interpreting, as opposed to translation or other interpreting modes.

Consecutive interpreting, for instance, often requires notetaking, which can result in a more structured output with closer surface-level lexical proximity to the original, but also heavy condensation of the source whenever possible. In the simultaneous mode different strategies affecting the surface level of the output are used, including summarisation, generalisation and expansion. This, in turn, requires fine-tuning of the existing metrics – a process that has been shown to work quite successfully compared to generalised MT metrics (Wein et al., 2024). However, the diversity of sub-genres in interpreting makes developing a comprehensive evaluation model for interpreting data extremely challenging. Studies further demonstrate that evaluation outcomes (e.g., BLEU) vary considerably depending on whether references are translations or interpretations, indicating limited cross-genre validity (Zhao et al., 2021). More broadly, recent studies on the robustness of MT evaluation metrics show that metric behaviour varies by domains and conditions, reinforcing the need for caution against assuming genre-invariant validity when applying automated metrics developed for translation to interpreting (Downie & Moorkens, 2026).

https://doi.org/10.26034/cm.jostrans.2026.8425

Note that while MT typically deals with written texts where fidelity to lexical and syntactic structure is measurable and often expected, interpreting is a spontaneous, spoken delivery of content across languages that prioritises communicative intent and real-time processing over literal lexical matching (Gile, 1995). As a result, interpreters' output is often so heavily reordered and compressed or expanded that it becomes much less amenable to metrics that depend on n-gram overlap with a reference, not to mention word error rate (WER) approaches that reward word-for-word correspondence. The application of BLEU and related metrics to interpreting is therefore limited. These tools systematically underestimate quality due to their insensitivity to semantic equivalence when phrasing diverges (Zhao et al., 2021). The semantic-lexical disconnection results in low BLEU/METEOR scores for interpretations that human raters may judge as successful or even excellent. From a pragmatic standpoint, this explains why there have been successful applications of automated metrics in MT evaluation, while in the context of interpreting there is no comprehensive and well-established automated metric – only more or less successful attempts.

Effective evaluation of interpreting requires metrics that account for semantic fidelity, pragmatic adequacy, and, crucially, communicative success – dimensions that cannot be captured by word-level overlap measures. This has been reflected in existing approaches to evaluating interpreting quality which mirror the evolving theoretical and methodological orientations in the field. Gerver (1969) emphasised the role of cognitive processing and the challenges inherent in simultaneous interpreting. Barik (1971) introduced a taxonomy of errors—including omissions, additions, and substitutions—that provided an empirical foundation for quality assessment. Pöchhacker (2001) and Kalina (2002) expanded further on these models, emphasising communicative effectiveness and listener orientation.

Among these, error-based evaluation remains one of the most operationalisable and widely adopted approaches. The method entails systematic identification and categorisation of deviations from the source text, allowing for both quantitative and qualitative analyses of interpreter performance. Such frameworks are particularly valuable for training purposes and for comparative evaluations across interpreters and settings. However, it is important to bear in mind that reception-based approaches are needed to complement error-based analyses if the evaluation is regarded as comprehensive and well-rooted in the pragmatic context (Kurz, 2001).

In light of the above, our work aims to explore the (semi)-automation potential of LLMs in error-based interpreting data evaluation. Given the limitations of MT-based metrics, an LLM-aided approach could partially automate the evaluation process and accelerate tasks that have so far been performed by human evaluators, such as detecting errors of different severity and nuanced meaning. This represents an important research gap that the present paper aims to address.

2.1 Error-based methods and the NTR Approach

One particular model of source-target text evaluation that merits attention in the context of error-based accuracy assessment is the NER model developed by Romero-Fresco and Martínez (2015), widely applied in the assessment of live subtitling. The NER model, originally introduced by Romero-Fresco (2011) and later enhanced, serves as

https://doi.org/10.26034/cm.jostrans.2026.8425

an established framework for evaluating the accuracy of live subtitles in media and live event broadcasting contexts. The acronym NER stands for the total number of words in live subtitles (N), the number of edition errors (E), and the number of recognition errors (R). In order to determine the proportion of accurate content, the values of E and R are deducted from N, and the result is divided by N. Further, the NTR model, proposed by Romero-Fresco and Pöchhacker (2017), expands upon the NER model and is specifically designed to evaluate interlingual respeaking. In the NTR model, translation errors (T) replace edition errors to assess the accuracy of the interlingual rendition.

Within the NTR framework, translation errors include omissions, additions, and substitutions (content errors), alongside concerns regarding correctness and style (form errors). Importantly, while the two models differ in terminology, both NER and NTR apply an identical system for grading error severity, assigning penalties of 0.25 points for minor errors, 0.5 points for standard/major errors, and 1 point for serious/critical errors. The NTR has been used beyond the narrow context of respeaking also to evaluate interpreting data. For instance, Korybski et al. (2022) applied the NTR model to compare interpreter output and output in a cascaded hybrid live communication workflow involving a respeaker and Machine Translation (MT). Similarly, Rodríguez-González (2024) used the model to investigate the extent to which Automatic Speech Recognition aids interpreters in delivering their work. To conduct NTR-based error analysis, he applied a custom-designed assessment grid based on the NER score spreadsheet used by Canadian media organisations for evaluating intralingual respeaking (Davitti & Sandrelli, 2020). The tailored grid enables segmentation of the source material, alignment of source and target idea units, and the evaluation of errors (both content- and form-related). This ensures a structured approach to data evaluation: first, the evaluators consider aligned sentence pairs and identify errors, often providing comments in the right-hand part of the grid. Subsequently, they input the perceived weight of the errors on the left-hand side of the grid to facilitate the final error score calculation. The advantage of the NTR scoresheet is its provision of a comprehensive and quantifiable error-based analysis. Particular error categories can be considered separately depending on the researcher's needs, and the evaluation of error severity allows for a more nuanced interpretation of the scores. For example, based on a large pool of bilingual data from interlingual respeakers, Davitti and Wallinheimo (2025) argue that omissions function as a statistically significant predictor of overall performance – a finding we applied in the current study to narrow down the analysis in the LLM-aided approach.

However, the NTR has significant drawbacks when applied to large sets of linguistic data: it is time-consuming and requires expert annotators as well as professional training to ensure consistency. Moreover, subjectivity and variability in judgments persist despite training. Another challenge associated with the NTR's application is its limited scalability due to heavy resource loading. Against this background, the current rapid technological changes in the language industry call for improved scrutiny of interpreting and, possibly, machine-supported approaches that can accelerate the process.

https://doi.org/10.26034/cm.jostrans.2026.8425

2.2 The Current Project

In the present study, we used the NTR approach, drawing on previous literature suggesting its high effectiveness in assessing interpreting outputs. Our main purpose was to provide a comprehensive assessment of interpreting errors affecting accuracy: omissions, additions and substitutions. This involved a stepwise procedure described in Section 3. Apart from that, we compared human-made assessments with those produced by ChatGPT, based on the assumption that technology could improve the evaluation process in at least two areas: the time required to conduct the evaluation and the consistency of the results. Here, we first focused on the most prevalent and predictive error type (i.e., omissions). Our overarching goal was to address the following research questions:

1) Can an LLM model produce similar results to human-made evaluations when considering interpreting outputs?
2) Does the use of LLMs allow for greater consistency and scalability of interpreting output evaluations?

We used two approaches to address these research questions: chain-of-thought prompting and specialised LLM agent training. We then compared the outputs produced by the LLM with those generated by human evaluators in terms of error number and category overlap, deviations from human-produced patterns and consistency of error detection. We wanted to establish if an LLM can perform as an accurate and reliable detector of major error categories, possibly highly consistent and less prone to overlooking error instances than human evaluators.

3. Method

In the present study, we utilised transcripts of interpreter outputs drawn from 40 speeches interpreted for a larger project originally designed with a different research focus. In that study, professional (n=27) and trainee interpreters (n=23) each interpreted five short (5-minute) speeches and then evaluated the speaker and speech difficulty. The speeches addressed general topics, such as hobbies, wellness, leisure activities, work-life balance, and had a similar word count and a mean Gunning Fog Index of 9.41 (Gunning, 1952). The study was designed to direct interpreters' attention to the speaker's accent rather than to lexical difficulty or grammatical structure. The interpreting outputs produced by the participants were recorded in audio form to enable further analysis (see Broś et al., 2025). The texts used in the original study were short and focused on general topics, which does constrain the generalisability of the findings. We did not test specialised vocabulary, more technical genres or longer interpreting tasks, leaving this for future studies. Crucially, however, the goal of this study was to compare two evaluation workflows under controlled and directly comparable conditions, rather than to benchmark LLM performance across domains or task difficulty levels. In this respect, the use of short, self-contained, clearly structured general texts is a strength rather than a limitation, as it ensures a stable "even playing field" where both human evaluators and the LLM operate on identical input, without introducing unnecessary complexity or noise.

Recognising the latent potential of these data in the current technological context, we redirected our attention toward quality evaluation. We therefore preselected 60 outputs

https://doi.org/10.26034/cm.jostrans.2026.8425

based on the participant surveys conducted in Broś et al.'s study, which identified the speeches that were the most challenging and the easiest to interpret. We selected these two speeches along with the one that obtained medium overall scores to ensure a balanced sample. Ten participants from each group were randomly selected, and the same three speeches they each interpreted were chosen, giving a total of 60 audio recordings (10 participants x 2 groups x 3 speeches). The audio data were subsequently transcribed using Whisper (Radford et al., 2023), then checked by human editors to eliminate ASR-driven transcription errors and subsequently realigned according to the preparatory procedure employed in the NTR model. A group of evaluators was trained to analyse the transcripts using a structured methodology followed by the design of two ChatGPT 4o (OpenAI 2025) evaluation approaches, as described in the following subsections.

3.1 Step 1. Human evaluation using NTR

The first part of the assessment process involved working with the tailored NTR grid described in Section 2.1 (with permission from Professor Elena Davitti of the Centre for Translation Studies, University of Surrey, UK). As we used interpreting data, we disregarded the part of the assessment grid that captures recognition-related errors and focused on the three error categories identified as those relevant for interpreting assessment (Barik, 1971): omissions, additions and substitutions. A sample file showing the grid is provided in Appendix 1.

Although the grid is not difficult to apply, it certainly requires training and practice, which is why we organised training sessions for two evaluators (both holding BAs in Applied Linguistics with C2-level English proficiency and native speakers of Polish who had attended interpreter training amounting to 60 hours at the time of project implementation). As the employment of novice interpreters for the evaluation of interpreting data can be considered a limitation in this type of investigation, we addressed this issue by organising introductory training in which examples of interpreter outputs were thoroughly discussed, with particular emphasis on the differences between approaches to assessing translation and interpreting outputs. The training was led by an active conference interpreter with extensive experience in applying the model. The interpreter was also available for consultation with the evaluators in case of any doubts or questions regarding the application of the grid and the evaluated material, as well as during the reconciliation phase. This approach ensured that interpreter expertise was integrated throughout the evaluation process. The training sessions lasted six hours in total and were complemented with reconciliation of practice materials to streamline the subsequent evaluation process. An important part of the training involved showing the evaluators how to align the source and target transcripts to ensure fidelity in comparison. Good alignment is the foundation of efficient evaluation because segments of interpreted output can be condensed or expanded to the point that they no longer correspond one-to-one with source segments. The NTR scoresheet applied in other large-scale research, such as the SMART project (Davitti and Wallinheimo, 2025), guides evaluators in identifying and annotating errors, assigning scores based on the error type (e.g., semantic, syntactic, pragmatic) and relevance (e.g., critical, minor, negligible) to the overall communicative success. The structured format enhances inter-rater reliability and facilitates analyses with different focal points.

https://doi.org/10.26034/cm.jostrans.2026.8425

Regarding inter-rater reliability, the agreement in error detection/overlap was assessed on a random sample of 100 sentences drawn from the evaluations produced by Evaluator 1 and Evaluator 2. Because this was an open annotation task involving rater subjectivity, the raters were not selecting from a fixed list of predefined error slots, and we were interested in the pre-alignment degree of precision and overlap in the errors detected by the two raters. For this reason, we used a metric based on precision and recall, i.e., the F_1 score. For the sample investigated, precision was 0.82, while recall (overlap) was 0.90, yielding an overall F_1 score of 0.86. Although not perfect, this score indicated strong inter-rater agreement in error detection, supporting the conclusion that the raters were largely consistent in their identification of interpreting errors across the sampled sentences, and that the data obtained were suitable for the reconciliation phase, in which the discrepancies were subsequently addressed. Reconciliation was a step in the process where the two evaluators and the interpreter met to discuss any discrepancies in their respective evaluation outputs e.g., misalignment of error categories or severity weights. Through discussion, the evaluators reached a final verdict on the score in these borderline cases, often adding qualitative comments to the grid to justify their decisions.

Using this approach, we obtained 14 scoresheets (one for each preselected participant), each with three tabs for the three texts separately, containing errors detected in the categories of omission, addition and substitution, weighted according to the scale proposed by Romero-Fresco and Pöchhacker (2017). This formed a human-driven assessment benchmark for our later comparisons with AI-driven evaluations of our source and target text pairs. The table summarising the number of errors in each category per text is provided in Appendix 2.

3.2 Step 2. Comparing Human Evaluation with LLM prompting outputs

In the subsequent phase of our study, we compared the outcomes of human evaluation with those generated by an AI-driven tool capable of processing large volumes of interpreting data (ChatGPT 4o). In line with the exploratory nature of the study, we intended to investigate whether an LLM can be effectively prompted to detect common interpreting errors and to replicate human judgment patterns. In order to track the performance of ChatGPT 4o in this task, we decided to first focus on one error category, omissions. The choice was driven by the fact that omissions are the predominant error category that affects interpreting performance. This is based on the commonsense premise that a good interpretation must capture the key content of the source text. Moreover, previous research (Davitti and Wallinheimo, 2025) has shown that omissions are a reliable and statistically significant predictor of overall interpreter performance: omission scores correlate with NTR scores.

When evaluating the quality of a target text in comparison to its source counterpart across different natural languages, various prompting strategies can be employed with LLMs, each reflecting distinct methodological priorities (Sahoo et al., 2025). Direct prompting involves instructing the model to assess the translation either holistically or based on predefined criteria such as adequacy, fluency, or fidelity. This is often achieved by requesting an overall quality score or justification, after providing the model with both the source and the target for comparison. Another possibility is comparative prompting: both source and target texts are presented to the model and evaluated side-by-side, which typically involves identification of divergences,

https://doi.org/10.26034/cm.jostrans.2026.8425

mistranslations, or other error types. In annotation-based prompting, the model annotates specific segments of the target text in relation to the source, highlighting translation errors or stylistic mismatches. Another approach, criteria-specific prompting, guides the model to focus on particular dimensions of quality such as lexical choice, syntactic accuracy, or pragmatic equivalence—using granular evaluation rubrics aligned with translation assessment frameworks (e.g., MQM or NTR). Prompts can also be based on references for the LLM – either reference translations or reference evaluations.

In addition to these structural approaches, prompting strategies can also vary according to the degree of prior examples provided to the model. In *zero-shot prompting*, the model is asked to perform a quality assessment task without being given any specific examples beforehand, relying solely on the clarity of the instruction and its pretraining. In contrast, *few-shot prompting* involves supplying the model with a small number of annotated examples of high- and low-quality translations, often including justifications or quality scores. This technique enables the model to infer task expectations and emulate expert judgment, improving alignment with human evaluative standards. Few-shot prompting is particularly useful when applying established error typologies or when aiming for consistent error categorisation across multiple evaluations. The choice between zero-shot and few-shot prompting significantly influences the reliability and granularity of the LLM's output and should be guided by the goals of the evaluation, its complexity, and the required level of interpretability (Sahoo et al., 2024).

Our first attempts at obtaining LLM-driven quality evaluations were based on a zero-shot comparative prompting approach, followed by agent training (see Section 3.3). Zero-shot prompting was justified methodologically in our project as it minimised prompt-induced bias and isolated the model's intrinsic translation competence, allowing evaluation without task-specific priming. A zero-shot approach also reduced confounding variables related to instruction framing. Moreover, zero-shot conditions approximate real-world, unsupervised deployment scenarios, thereby supporting ecological validity. Consequently, the approach was intended to facilitate a cleaner assessment of adequacy, fluency, and semantic fidelity based on the model's representations. However, given problems with batch processing and the need to enter transcripts into the chat interface one by one, which considerably increased the workload, we decided to reduce the initial dataset to 40 texts from the initial 60 texts. This is described in detail in section 4.1, which includes results for the final dataset of those 40 texts.

3.3 Step 3. Comparing Human Evaluation with LLM agent outputs

In our agent construction strategy, we drew from our earlier prompting attempts as described in 4.1, after which we found that, firstly, ChatGPT tended to hallucinate. Secondly, it tended to confuse omissions with substitutions, and thirdly, it was not consistent in repeating precisely the same procedure across more than one pair of texts. Thus, when creating instructions for the agent, we followed a set of general principles described below. Also, to streamline work with multiple comparisons between ChatGPT and human outputs, we decided to use a subset of six texts from two participants when testing the agent.

https://doi.org/10.26034/cm.jostrans.2026.8425

3.3.1 Agent training details

We did not intend to request a specific result without providing the agent with its role, context, task details, and the desired output. We treated ChatGPT as an employee, defining its specific role as an expert evaluator of interpreting performance. We specifically emphasised that it must not hallucinate and that if it was not sure what to do, it should ask or consult its training materials. We then defined the context of its task and added training materials which initially included one pairing of a source and target text presenting errors detected by a human evaluator together with the scores. We explained the three error types and their three levels of severity, providing examples of each. We included all error types to avoid ChatGPT mistaking other phenomena for omissions. We also noted that more than one error can occur in a given sentence, in which case all errors should be reported. Apart from that, we instructed the agent not to consider sentences marked as n/t (*not translated*, major omissions) in the target text. We then described the task as a sequence of actions to be carried out in the specified order. At the end of the evaluation process, ChatGPT was asked to create a table based on an example containing only column headers, which was provided in the training materials. The table was to be populated with the detected errors, their types (label), and severity. The agent was instructed to save it as an Excel file.

After preparing the detailed instructions and training files, we proceeded to prompt the agent and correct initial errors and inconsistencies. We asked ChatGPT to explain why certain errors occurred and adjusted the instructions accordingly. The input file was initially provided as a table in a Word document. At this stage, major issues included difficulties in decoding files, saving the output in the correct format and failing to analyse sentences beyond the first three to five that were subjected to preliminary screening by the chatbot.

The initial interaction with the agent was promising, with one text evaluated in close alignment with the human assessment. However, as more texts were introduced, problems emerged, leading us to explore different prompting strategies to address them. The details of these attempts are provided in Section 4.2.

4. Results

Given our stepwise approach and the use of human evaluation as a benchmark for LLM performance assessment, we present the results descriptively. Working with ChatGPT involved multiple revisions of the prompts as intermediate results were not satisfactory. We therefore outline the major problems encountered and our tentative solutions, which, on the one hand, contributed to slight improvements, and, on the other, resulted in deterioration. We illustrate these attempts and adjustments in figures and tables to enable a better understanding of the process. The original outputs were in Polish; for greater clarity we focus on the English-language explanations provided by the LLM, accompanied by literal translations of the content.

4.1 Prompting procedure results

As stated in Section 3.2, we started our prompting attempts with the *zero-shot* approach. The figure below presents the prompting sequence applied in the first

https://doi.org/10.26034/cm.jostrans.2026.8425

automated analysis trial implemented to examine whether automation in this regard was feasible and how close its performance was to human evaluation:

ZERO-SHOT PROMPTING SEQUENCE FOR FEASIBILITY TESTING

The sequence below presents the exact wording of the prompts used to elicit omission detection and description in the investigated data set

1. I want to compare Text 1 and Text 2. Text 1 is in English and it is a transcript of a source speech. Text 2 is in Polish and it is a transcript of live interpretation of the source speech. I want to evaluate the quality of the interpretation (Text 2). Suggest both quantitative and qualitative ways to assess the quality of Text 2. n/t in the target text means an untranslated fragment. Here is Text 1 (source text): […]; here is Text 2: […]
   [Reaction: ChatGPT describes some possible evaluation approaches including error-based approaches].

2. Please elaborate on error-based approaches and create an error-based framework for evaluation and provide examples from the text provided. [Reaction: ChatGPT produces a more granular error-based evaluation plan with concrete error examples, including errors of omission]
3. Now focus on the errors of omission and create an annotation table just for these errors based on the source and target texts. Include n/t fragments (untranslated content) in your analysis. Grade the errors for severity as 'minor' or 'major'. Include a column with a brief comment explaining why you think the line contains an omission error. Quite only the lines from source and target where there is content omission, including the lines marked n/t. [Reaction: ChatGPT produces the table with a list of sentence pairs and the requested explanations/comments].

Figure 1. Initial zero-shot prompting sequence.

At this stage, we obtained automated analyses that indicated a promising level of concordance between human and AI assessments in the error category of omissions. Again, we used F_1 to assess precision and overlap (recall), and the results ranged from 0.72 to 0.90 (depending on the text), indicating a moderate to strong overlap between human and LLM-made lists of omissions, as shown in the table below for one example.

Source sentence	**Aligned target sentence detected by the LLM as an erroneous sentence**	**LLM comment**
We may not be able to afford the good food and medical care we need.	Może nas nie być stać na opiekę zdrowotnej, której potrzebujemy. Backtranslation: '*We may not be able to afford the heath care we need*'.	*Minor error; 'Good food' omitted. Relates to physical wellness.*

Figure 2. LLM output fragment.

However, F_1 scores are not granular enough to reveal the discrepancies resulting from pragmatic nuance and contextual appropriateness—areas in which human evaluators exhibited greater sensitivity. Consider the example provided below: despite overlapping error detection, the weighting of the error assigned by the LLM differed from the human assessment. The human evaluator identified the error as major

https://doi.org/10.26034/cm.jostrans.2026.8425

because it interfered with the logical structure of the fragment (the introductory phrase linking to the previous content, 'when this happens' is missing) and thus affected the recipient's understanding. ChatGPT, however, classified the error as minor, revealing subpar contextual awareness of the automated tool.

Source sentence	Aligned target sentence detected by the LLM as an erroneous sentence	LLM comment
When this happens, we may even question our own sense of meaning and purpose.	Możemy nawet kwestionować nasz sens życia. Backtranslation: *'We may even question our meaning of life'.*	Minor error: meaning omitted; Less critical as purpose and meaning overlap, but nuance lost.

Figure 3. Error severity according to ChatGPT (zero-shot prompting).

In the second step, we attempted to leverage the apparent benefits of using ChatGPT for error detection by scaling up the procedure and asking the model to process a bulk file containing 40 source-target text pairs, with a prompt added to the original sequence to aid the model in interpreting the text file correctly. The results of this approach were disappointing: despite numerous attempts to refine the prompt, the model was unable to produce analyses based on the uploaded bulk text files. The output featured random comments placed in unpredictable locations in the uploaded bulk text (re-edited by the LLM), as well as modifications and deletions of large fragments of both source and target, and even entire pages. This clearly indicated that there were technological constraints on the extent to which the procedure could be accelerated, and it also highlighted the continued importance of human scrutiny in any attempts to streamline language data analysis workflows with LLMs.

In the third step, we reverted to using ChatGPT's interface only rather than uploading text files. To facilitate task replication, we added the following steps to the initial prompting sequence.

1. Using the same approach, analyse another set of texts with a new source text. The new source text is here: […]; the new target text is here […]. Please export the table to a spreadsheet. [Reaction: ChatGPT produces the table and exports it to a downloadable XLS file].
2. Now consider a new source text and Target Text X. Repeat the omission error analysis procedure as with the previous source text. [Reaction: ChatGPT produces the table and exports it to a downloadable XLS file].

Figure 4. Additional steps in the prompting sequence.

Our intention in expanding the prompting sequence was to avoid one-by-one prompting of source-target text pairs. Given the number of texts we wanted to analyse, we hoped to expedite the procedure and assess the consistency of the LLM's performance. The outcomes of this approach, however, were again far from satisfactory: while the first text pair, analysed immediately after prompting the model, produced results similar to the initial feasibility checks, the repeated procedures proved problematic. It seemed as though the LLM had 'forgotten' the entire prompt and no longer adhered to the key task of analysing solely sentence pairs including omissions. Instead, the output included examples of 'good practice', i.e., sentence pairs in which

https://doi.org/10.26034/cm.jostrans.2026.8425

the interpreter omitted no content. Although potentially useful in different analytical approaches, this type of feedback from ChatGPT diverged from the original prompt and resulted in hallucinations.

Furthermore, we noticed a marked tendency for the LLM to reduce its output with further iterations of the task. In other words, in later analysis rounds prompted simply with "Repeat the omission error analysis procedure as with the previous source text.", the output gradually deteriorated in volume and precision and was more prone to hallucinations. The degradation of output quality was consistent: repeated attempts produced the same trend, leading to extreme cases where the overlap between human and AI evaluations dropped below 10%, thus rendering the application of LLMs with this procedural approach ineffective. Consider the following example to illustrate the above phenomenon: while there is a clear substitution error (concrete information about type 2 diabetes is replaced with a generic statement about being in the 'risk group'), ChatGPT makes an unexpected (and incorrect) comment that the error has 'no severity' and further erroneously claims that the 'health risk is correctly stated'.

Source sentence	Aligned target sentence detected by the LLM as an erroneous sentence	LLM comment
This means you may also develop type 2 diabetes.	'Oznacza to, że ty też możesz być w grupie ryzyka.' Backtranslation: 'This means you can also be in the *risk group*'.	No severity, health risk is correctly stated.

Figure 5. Error non-detection example.

We analysed the outputs of the one-by-one prompting batch in detail by comparing the model's outputs with the human evaluations from the NTR. As before, we calculated the overlap (recall) between the two by determining how many of the errors identified by human annotators were successfully detected by the LLM. This was then divided by the total number of errors (see table below). Errors detected by the LLM but not human evaluators were not counted.

Participant code	Text 1 overlap	Text 2 overlap	Text 3 overlap
P5	0.33	0.50	0.14
P6	0.19	0.31	0.18
P10	0.29	0.42	not evaluated
P11	0.53	0.11	0.54
P13	0.33	0.36	not evaluated
P17	0.42	0.25	0.85
P19	not evaluated	0.2	0.54
P20	not evaluated	not evaluated	0.72
PRO1	not evaluated	not evaluated	0.33
PRO4	0.31	0.2	0.1
PRO7	not evaluated	not evaluated	0.33
PRO8	not evaluated	not evaluated	0.5
PRO11	0.67	0.31	0.63
PRO15	0.67	0.63	0.71
PRO17	0	0.5	0.63
PRO20	0.55	0.29	not evaluated
PRO24	0.50	0.45	not evaluated
PRO28	0.43	0.33	not evaluated
Means	**0.4**	**0.35**	**0.48**

Table 1. Human – LLM error detection overlap.

https://doi.org/10.26034/cm.jostrans.2026.8425

Following the prompting phase of testing, we concluded that one viable approach to applying LLMs for error detection in interpreting output was to closely control the model, possibly performing analyses on a one-by-one basis always followed by human scrutiny. Although less attractive than bulk or repeated analyses, this approach saves a considerable amount of time compared to fully human-driven evaluation carried out from scratch.

Another option considered at this stage was the development of an LLM agent to ensure more robust and consistent analytical performance – this approach is described in Section 4.2.

4.2 LLM agent results

As described in Section 3.3, we initially trained an agent using one pair of model texts and provided it with one source-target text pair for evaluation. The results were promising: no sentences were omitted in the analysis, all types of errors were detected and the general overlap with human evaluations was 60% (0.6 score). The agent failed to detect six errors, which were usually in sentences including more than one type of error. There was also one hallucination. The results of the analysis, together with a comparative table of errors marked by human evaluators, are provided in Appendix 3.

In addition to listing errors, we asked ChatGPT to provide a general score for the text on a scale from 1 to 10. This proved confusing as it produced a separate score for each sentence. When prompted to correct its approach and evaluate the whole text, ChatGPT produced an error summary, assigned a score, and provided a justification:

https://doi.org/10.26034/cm.jostrans.2026.8425

Error Summary:

- **Omissions**: 10 (including 3 critical ones)
- **Substitutions**: 19 (with several major and one critical)
- **Additions**: 0 (no unjustified insertions detected)
- **Accurate renderings**: Very few fully accurate lines
- **n/t (untranslated)**: 6 lines completely untranslated

Scoring Logic:

A score of **10** would require an accurate, complete, and nuanced interpretation.
Here, the score must be reduced due to:

- Several critical errors
- Multiple omissions of key concepts
- Inconsistent rendering of core terminology (e.g., "wellness," "balance," "roles")
- Lack of cohesion in transferring the logic of argument from English to Polish

Final General Score: 4 out of 10

This reflects a basic level of comprehension and partial delivery of meaning, but with major gaps in completeness and fidelity to the original. There's significant room for improvement in both coverage and semantic accuracy.

Figure 6. Error scoring by ChatGPT

After this initial promising result, we decided to repeat the procedure with subsequent texts via batch processing. Nonetheless, the agent did not perform well this time. With more text pairs, ChatGPT started to omit sentences or marked all mistakes in the same category (e.g., only omissions). It also seemed to have problems with n/t sentences and file formatting. When asked to self-diagnose the problems and suggest solutions, it claimed that it "overly simplified the logic to rush through the batch; didn't adapt the same rigorous error detection strategies it used in single comparisons; its structure detection (for pair extraction) lacked robustness and its evaluation logic lacked nuance when handling multiple pairs simultaneously." The agent was able to state what it should do better next time, but it was then unable to implement these adjustments. Therefore, we reverted to feeding the agent with one pair of texts at a time. However, this did not meet with success. Every subsequent pair was burdened with progressively worsening performance. Our next training attempts included:

1. Working with different input files: tables in Word proved to be problematic in the long run. We switched to Excel tables which were refined until ChatGPT could process them correctly. After numerous unsuccessful attempts at obtaining consistent evaluations, we also tried working with texts that were not divided into sentence pairs, hoping that ChatGPT would process them as a whole and detect errors. This is described more thoroughly in point 3. The various types of input files used with the agent are provided in Appendix 4.

https://doi.org/10.26034/cm.jostrans.2026.8425

2. Adding more training files in the form of eight different examples of human evaluation. The agent was then prompted to consult its training files and report on what it had learned at the beginning of the conversation. The procedure provided insight into how the agent processed the training data, which was satisfactory but did not improve the agent's performance. The results of the self-report varied slightly each time the agent was asked but were similar in content (see Appendix 5 for an example).

3. Refining instructions to remedy particular problems.

First, one of the major problems we detected was that not all the lines/sentences from the input file were evaluated. When prompted to correct this, the agent did not improve. We concluded that the problem might have been related to the n/t sentences. When the agent corrected itself in a semi-satisfactory manner and was then asked to repeat the same procedure with a new file, it made the same mistakes, or experienced problems with the file formatting, which was unexpected since all of our input files were structured in the same way. It seems that the agent was easily distracted and inconsistent in its internal procedures despite identical instructions and its 'promises' to follow them.

Another issue that needs to be addressed is an apparent trade-off between the desired output format and error detection. The more detailed the error descriptions were, the more likely it was that some errors would go undetected. Conversely, reducing the level of detail led to more errors being detected, but with very generic comments, no error description at all, or a persistent focus on a single error type. As for our remedies, adding instructions such as "Evaluate all the lines from the file. Do not omit any errors. Do not invent errors that were not there". did not improve the agent's performance and seemed to contribute to its weaker performance. Instructing it to ignore n/t lines and focus on errors and comments resulted in another trade-off. The agent would either omit some sentences from the input or, when prompted to correct itself, forget the instruction to ignore the n/t lines. Thus, we redirected the instruction again to include n/t and avoided elaborate comments (earlier instruction included "Explain why you think the line contains that type of error. Be as specific as possible"). Consequently, the agent fixated on n/t lines and included only one error category in the evaluation, omissions.

Finally, we tried excluding n/t lines from the files altogether, making it easier for the agent to ignore them. In this case, the data could be analysed sentence by sentence without distractions and n/t omission scores could be added later by a human overseer. This change produced promising preliminary results. When prompted to explain how it performed the evaluation, the agent replied:

https://doi.org/10.26034/cm.jostrans.2026.8425

"I only logged sentences where I detected one or more of the [errors – listed]. Each of these errors was entered in the output table as a separate row, and in cases where multiple issues occurred in one sentence pair, each issue was listed separately.

🚫 What was not included:

- Sentences that were accurately and completely translated—these were not entered in the table because they contained no errors […].

If you want, I can also provide a table where:

- All sentence pairs are shown.
- An explicit "No error" label is added for correct translations."

Figure 7. Agent output after n/t line exclusion.

In light of the above, we asked the agent to create tables including correct translations, which it successfully produced. However, this came at a cost. The agent struggled to assess sentences containing multiple errors. It consistently identified a single error, usually the first, or none at all, whereas human raters detected two or three. No sentence with multiple errors was accurately evaluated, and this approach yielded the lowest overlap score among all our attempts.

Participant Code	Overlap score
P5_1	0.03
P5_2	0.12
P5_5	0.04
P6_1	0.15
P6_2	0.03
P6_5	0.03

Table 2. Overlap scores comparing the agent with human evaluators.

Another problem we detected was a gradual deterioration of severity judgments. We therefore skipped this step and limited the instructions to error type detection. No improvement was observed as a result. The agent continued to mark only n/t lines as omissions while failing to detect other errors. We then experimented with including n/t lines in the evaluation, but only after the initial error detection stage; this led to processing problems with some lines left evaluated or errors omitted. Subsequently, we tried to specify the steps more explicitly in the instructions, e.g., "detect how many lines there are in the file and then evaluate all of them one by one". We also experimented with reordering the training instructions; none of these changes produced the desired results. On the contrary, efforts to improve the instructions resulted in the chat making unpredictable mistakes elsewhere. Since the agent was now omitting large parts of the input files, as if it were failing to detect all the sentences, we asked it to address the problem and suggest revisions to the instructions. It suggested that the phrasing 'all rows must be processed in full' and 'each row present in the input file' should be included, as well as recommending some quality control (a consistency check across all rows to ensure uniform error classification). Thus, ChatGPT admitted that a final human check was necessary.

Nonetheless, although these 'improved' instructions were more precise and formulated in line with the chat's suggestions, they did not result in better evaluations; the same problems persisted. Furthermore, the changes exacerbated the problem by placing excessive emphasis on the training materials. Although the chat itself proposed the improvement to ensure that all types of errors would be detected, it instead caused the agent to focus on matching input texts to the training materials, worsening its performance.

https://doi.org/10.26034/cm.jostrans.2026.8425

After numerous attempts to detect and remedy individual issues, we decided to change our strategy. One of the main mistakes was that not all the sentences were processed, with some omitted from error detection and/or reporting. We concluded that the issue stemmed from our approach as we were asking ChatGPT to perform an unnatural task: treating texts as tables of sentence correspondents. We shifted to providing the agent with complete texts along with their interpretations instead of segmented tables, as ChatGPT is good at analysing, summarising and correcting texts. This approach yielded better results. However, the analysis then rested on larger chunks (segments of data) and the agent still failed to identify all omitted sentences, apparently ignoring some of them. When prompted, it would correct some omissions but neglect others. Repeated reminders led to partial corrections, performance slowed and the agent began reporting each step sequentially in the conversation. It continued to process the text in arbitrarily defined segments. When asked to streamline the procedure and perform the analysis internally, its performance declined leading to additional skipped errors. Performance also deteriorated over time when processing successive input files. After several attempts, the system stalled and had to be reactivated with new prompts, after which its performance degraded further. Overall, the number of undetected errors in the analysed samples ranged from 8 to 34. The overlap scores across the examined data sets were inconsistent (0.02-0.28), indicating variation in the alignment between human and machine error detection.

Participant Code	Overlap score
P5_1	0.2
P5_2	0.28
P5_5	0.11
P6_1	0.02
P6_2	0.26
P6_5	0.24

Table 3. Overlap scores based on the analysis of complete texts rather than sentence-by-sentence.

Due to frequent omissions of entire sentences by interpreters and the removal of n/t annotations from the input, discrepancies arose between the number of source and target sentences. Consequently, ChatGPT failed to process these texts properly. When not prompted to segment the text in a specific way, the chat divided it into clusters of 2-5 sentences, collectively matching them to a Polish equivalent and providing only one error category, as shown in the example below.

Clustered sentences in the output	An equivalent in Polish	Error category given by the agent
Wellness is a broad concept. Let us try to give it a definition. People usually have an intuitive sense of what wellness means.	Ludzie zazwyczaj czują czym jest zdrowie. [*People usually feel what health is.*]	Omission

Figure 8. Example of excessive simplification of evaluation by ChatGPT.

In this case, the human rater evaluated the three sentences individually, assigning the errors in the first two sentences (“Wellness is a broad concept” and “Let us try to give it a definition”) to the category “n/t” (not translated) as they had been omitted by an interpreter, and classifying the errors in the third sentence (“People usually have an intuitive sense of what wellness means”) into two categories, omission and substitution. The example shows that the agent’s analysis is factually incorrect, as it

https://doi.org/10.26034/cm.jostrans.2026.8425

distorts the actual number of errors, is less rigorous than that of the human rater, and does not do justice to the true quality of the input.

Another problem with this approach was that the agent acted as if it were operating autonomously and interfered with the input. In one such case, the original sentence read “In these times, our habits and routines can help us get that feeling of control back”. In the output, the agent intervened and truncated the prepositional phrase “in these times”, thereby generating and evaluating the incomplete sentence “Our habits and routines can help us get that feeling of control back.”

Yet another independent action taken by the agent relates to content fabrication. Instead of limiting itself to evaluating sentences in the input as specified in the prompt, the agent took the initiative to generate new ones, e.g., the following sentence: “Volunteering helps solve problems, strengthen communities, improve lives, connect to others, and transform our own lives” did not appear in the source texts. In doing this, the agent may have relied on the contextual cues and may have been sensitive to the semantic content and sentence structure of the input. The data in the table below indicate an increase in the number of sentences fabricated by the LLM – exceeding the number of source sentences and thus impacting the evaluation.

Sentences fabricated by the agent	Source sentences
Volunteering *helps solve problems, strengthen communities, improve lives, connect to others, and transform our own lives.*	**Volunteering** is about giving, contributing, and helping other individuals and the community at large.
Volunteering *also provides an opportunity to practice important skills used in the workplace...*	**Volunteering** doesn’t have to involve a long-term commitment or take a huge amount of time out of your busy day.
Volunteering *can help take your mind off your worries, keep you mentally stimulated, and add more zest to your life.*	**Volunteering** connects you to other people.
Volunteering *can be a stepping stone to a new job or career...*	
Volunteering *can help you make new friends, expand your network, and boost your social skills.*	
Volunteering *can also help you build upon skills you already have and use them to benefit the greater community.*	

Figure 9. Hallucinations by the agent leading to inflated sentence count.

Finally, the examples below show that the agent clustered the English source sentences and applied various strategies in response to the same prompt. These ranged from inventing a sentence in the absence of a Polish equivalent (a), generating an equivalent despite the presence of a source sentence (b), adopting a combined approach — substituting a sentence in Polish with the one created by the agent and generating sentences in the absence of source sentences (c), to replacing a shortened interpretation with a more complete rendition that offered greater syntactic and lexical detail (d). In some cases, e.g., (b) and (c), the generated sentences were semantically incongruent with the source.

https://doi.org/10.26034/cm.jostrans.2026.8425

Source sentences in English	Source sentences in Polish	Sentences in Polish invented by the agent
a) All wellness dimensions are related to one another. We can say that one builds on another.	n/t [not translated]	Wszystko w naszym życiu jest ze sobą połączone.
b) This may be caused, for example, by debt or not being able to afford what we need, which can lead to medical problems and trouble at work.	Na przykład boimy się, że nie spłacimy długu. To może prowadzić do różnych problemów, także w pracy.	Pojawia się w naszym życiu niepokój, który może przerodzić się w depresję.
c) Medical problems mean a lack of physical health. Questioning your purpose relates to spiritual health. These are closely related here.	Może to prowadzić również do problemów ze zdrowiem, problemów natury medycznej. [*It can also lead to health problems, medical problems*.] n/t [not translated] n/t [not translated]	Problemy fizyczne wpływają na nasz stan zdrowia. A problemy duchowe powodują problemy z poczuciem celu w życiu. [*Physical problems affect our health. And spiritual problems cause problems with our sense of purpose in life*.]
d) We are students, friends, parents, spouses, coworkers, hobbyists, community members, and citizens.	Pełnimy w życiu wiele ról.	Jesteśmy rodzicami, współpracownikami, znajomymi, członkami społeczności.

Figure 10. Divergent agent strategies in evaluation.

Given the results of the approaches discussed above, we concluded that the problems ChatGPT encountered in the evaluation task were not solely due to the wording of the prompts but also stemmed from its internal inconsistency and algorithmic limitations. These issues are further addressed in Section 5.

5. Discussion and Conclusions

As demonstrated by the results of our analyses, LLMs such as ChatGPT are a promising tool for automating interpreting quality assessment — with some important caveats. While our study focused on the predominant error category of omissions, LLMs can also detect other error categories, such as substitutions and additions. They can further provide a general (qualitative) assessment of interpreting quality and determine the severity of detected errors. Furthermore, there is potential in their ability to consistently detect the same errors across texts and find errors not detected by human evaluators. However useful, these affordances are nonetheless limited in relation to the task at hand. Since the key goal of using such tools is to batch-process larger amounts of data, their unpredictable behaviour and deteriorating error-detection performance are particularly problematic. The mean recall score across the batch tasks that we administered did not exceed 0.18 (SD=0.13), with as little as 2% of overlap between ChatGPT and human evaluations in some cases, as calculated based on the six texts produced by two different interpreters that were used both in the prompting phase and in agent training. Comparing the most successful strategies applied in this diagnostic study (see Tables 1-3), we conclude that batch prompting

https://doi.org/10.26034/cm.jostrans.2026.8425

yielded the best results (mean F_1 score 0.28, SD=0.13, with up to 50% overlap in one case). Agent training resulted in worse results, with a mean of 0.07 (SD=0.05) in the case of sentence-by-sentence work by our agent, and a mean of 0.19 (SD=0.1) when the agent was asked to process whole texts. Thus, overall, the results of our attempts to batch-process error detection were far from satisfactory.

Multiple issues arose during both prompting and agent training. The chat's behaviour was often erratic, and any improvements achieved through self-correction, human correction or additional prompting were short-lived. ChatGPT proved inconsistent in following instructions and in repeatedly applying the same procedure. Moreover, it often failed to complete an evaluation, breaking off after the initial part of the analysed file or selectively excluding certain error types or lines of text. It is worth noting that ChatGPT would sometimes ask whether it should continue after certain steps, while at other times it would proceed independently. It is also inconsistent in how it reports its own processing. It often evaluates the same texts differently, which affects the presentation of the output and worsens performance, resulting in inconsistent feedback on its own operation. This makes its behaviour unpredictable, and, without time-consuming human supervision, there is no guarantee of its long-term reliability.

Perhaps the most disappointing aspect of ChatGPT's performance in this diagnostic study is that there seems to be no reliable remedy for the problems that arise. Even small changes to the instructions can lead to unpredictable behavioural shifts that are not directly related to the changes introduced in instructions. Thus, improving one aspect of the model's performance often results in substantial deterioration in other parts of the task. Consequently, there is no feasible strategy that would reliably lead to an optimal agent or prompt at this point. We attribute this problem to the way LLMs are trained and structured rather than to the task itself or to our particular instructions. Since any change in the instructions aimed at optimisation produced worse results and since we intended to remedy specific problems carefully through trial and error, removing one variable at a time, we conclude that at the time of the study (2025), the tested LLM was not yet ready to deliver fully-automated and comprehensive error detection in batches of interpreting outputs. To address the shortcomings in LLM-driven evaluation, we modified the chat's task by changing only one element at a time: 1) inclusion vs. exclusion of n/t sentences from evaluation; 2) inclusion vs. exclusion of error severity judgments; 3) use of Word vs. Excel files; 4) sentence-by-sentence tables vs. whole texts; 5) prompting the model to provide detailed error descriptions vs. omitting such instructions, among others. All these attempts were conducted separately to ensure that specific issues were solved. However, after corrective prompting the LLMs improved in one respect, but this generated other issues, often more severe than the original problem. These findings are consistent with prior work on prompt sensitivity (Reynolds & McDonell, 2021; Webson & Pavlick, 2022) and on instability in model outputs under minor prompt perturbations (Wang et al., 2022). Related research on long-context models also shows performance degradation when relevant information is distributed across extended inputs (Liu et al., 2024; Li et al., 2025). This suggests that LLM behaviour can be highly sensitive to prompt formulation and input structure, which makes reliable optimisation in complex, multi-step evaluation tasks difficult to achieve reliably. Furthermore, fine-tuning prompts changes the way models interpret them. Seemingly equivalent prompts can yield different performance—even if a task was previously executed well (Kotha et al., 2024).

https://doi.org/10.26034/cm.jostrans.2026.8425

Returning to our research questions, we cannot confirm that at the time of its application, the LLM model we used produced results comparable to those of human evaluations. While it showed promise in processing the first iteration of evaluation following a precise prompt, its capabilities were not sufficient to address the task consistently when repeated performance was requested or when text files were uploaded to the chat for processing. This means that substantial human scrutiny is required when combining AI and human evaluations in larger projects where large amounts of data must be analysed. However, potential benefits of LLMs in interpreting quality assessment exist and are worth emphasising. First, although we have shown that ChatGPT fails to replicate human results in repeated evaluation, our findings also demonstrate that a closely supervised procedure can accelerate data analysis – it seems fair to expect a shortening of the duration of human-led evaluation after an initial round of LLM-driven error detection triggered by a precise and tested prompt. While human supervision remains necessary, the model produces outputs that can serve as a useful foundation for more granular, human-driven analysis, particularly under well-structured zero-shot prompting. Secondly, in our first, exploratory study, Chat GPT 4o, when 'freshly prompted', was able to identify some error instances that were undetected by human evaluators, despite the multi-step evaluation process involving reconciliation. Evaluating bilingual data with the NTR model is a very tedious task that demands a high level of attention to detail, carries the risk of human misjudgement, and makes instances of error non-detection seem inevitable. LLMs can help address this issue when appropriately controlled, reducing human effort and, possibly, offering a way to better manage resource loading in evaluation projects. Thirdly, the application of the model exposed some clear shortcomings of the NTR evaluation approach: there are some borderline cases where, for example, errors can be assigned to more than one category (e.g., a substitution that causes a partial loss of content) or weight (the decision whether an error is 'minor' or 'major' is often subjective). Against this backdrop, the LLM used in the present study seemed to seek a more 'binary' and consistent approach, which may work well when analysing large bilingual data sets. Finally, as a result of applying the model, we realised that focusing on one error category carries with it an intrinsic constraint, as errors can be interconnected in some sentences, and ChatGPT seemed to be able to detect that interconnection, thus delivering an evaluation that was broader than requested, but closer to the actual way languages work.

In terms of methodological recommendations based on our work, future studies employing LLMs for analytical purposes should consider several procedural refinements. These include avoiding plain-text file inputs in favour of more structured data formats, maintaining a minimal degree of abstraction between the prompt and the model's output, and systematically verifying each output against the intended analytical criteria. Periodic re-administration of the prompt — or explicit elicitation of the model's interpretation thereof — may further enhance consistency and reliability. It is also recommended that any LLM-generated analysis be subject to a final stage of human-led validation and approval prior to incorporation into research findings.

Several avenues remain open for future investigation. First, it would be valuable to examine the extent to which LLM-based evaluations encompassing broader linguistic dimensions — such as style, register, and grammatical accuracy — align with human judgements, given that error analysis and formal accuracy represent only one facet of translation or interpreting quality. Secondly, further research is needed to quantify the

https://doi.org/10.26034/cm.jostrans.2026.8425

time savings achievable in LLM-assisted interpreting evaluation relative to fully human-driven analytical approaches. Finally, given that successive iterations of LLMs can be expected to demonstrate progressively greater proficiency in responding to prompts and generating reliable evaluations, it is essential that future research projects engage with the most current model versions available at the time of investigation, so as to ensure that findings reflect the state of the art in automated analytical capability.

References

Barik, H. C. (1971). A description of various types of omissions, additions, and errors of translation encountered in simultaneous interpretation. *Meta, 16*(4), 199–210. https://doi.org/10.7202/001972ar

Banerjee, S., & Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization*, 65–72. https://aclanthology.org/W05-0909.pdf

Broś, K., Czarnocka-Gołębiewska, K., Mołczanow, J., & Szupica-Pyrzanowska, M. (2025). The role of expertise in coping with accents during simultaneous interpreting. *Interpreting, 27*(1), 52–86. https://doi.org/10.1075/intp.00117.bro

Davitti, E., Korybski, T., Orăsan, C., & Braun, S. (2025). Quality-related aspects. In E. Davitti, T. Korybski, & S. Braun (Eds.), *Routledge Handbook of Interpreting, Technology and AI* (pp. 305–326). Routledge. https://doi.org/10.4324/9781003053248

Davitti, E., & Wallinheimo, A.-S. (2025). Investigating cognitive and interpersonal factors in hybrid human–AI practices: An empirical exploration of interlingual respeaking. *Target. International Journal of Translation Studies, 37*(2), 244–270. https://doi.org/10.1075/target.00035.dav

Downie, J., & Moorkens, J. (2026). What do the metrics mean? A critical analysis of the use of automated evaluation metrics in interpreting. *arXiv:2601.05864*. https://doi.org/10.48550/arXiv.2601.05864

European Commission Directorate-General for Interpretation (SCIC). (n.d.). *Consecutive and simultaneous interpreting marking guidelines*. https://europa.eu/interpretation/doc/marking_criteria_en.pdf

Gerver, D. (1969). The effects of source language presentation rate on the performance of simultaneous conference interpreters. In F. Pöchhacker & M. Shlesinger (Eds.), *The Interpreting Studies Reader* (pp. 52–66). Routledge.

Gile, D. (1995). *Basic concepts and models for interpreter and translator training*. John Benjamins. https://doi.org/10.1075/btl.8

Gunning, R. (1952). *The technique of clear writing*. McGraw-Hill.

He, H., Boyd-Graber, J., & Daumé III, H. (2016). Interpretese vs. translationese: The uniqueness of human strategies in simultaneous interpretation. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 971–976. https://www.cs.umd.edu/~jbg//docs/2016_naacl_interpretese.pdf

Kalina, S. (2002). Quality in interpreting and its prerequisites – A framework for a comprehensive view. In G. Garzone & M. Viezzi (Eds.), *Interpreting in the 21st Century: Proceedings of the 1st Conference on Interpreting Studies, Forlì* (pp. 121–130). John Benjamins. https://doi.org/10.1075/btl.43.12kal

Korybski, T., Davitti, E., Orăsan, C., & Braun, S. (2022). A semi-automated live interlingual communication workflow featuring intralingual respeaking: Evaluation and benchmarking. *Proceedings*

https://doi.org/10.26034/cm.jostrans.2026.8425

of the Thirteenth Language Resources and Evaluation Conference, 4405–4413. European Language Resources Association. https://doi.org/10.18653/v1/2022.lrec-1.468

Kotha, S., Mitchell Springer, J., & Raghunathan, A. (2024). *Understanding catastrophic forgetting in language models via implicit inference* [Conference poster]. 12th International Conference on Learning Representations (ICLR 2024), Vienna, Austria. https://openreview.net/forum?id=VrHiF2hsrm

Kurz, I. (2001). Conference interpreting: Quality in the ears of the user. *Meta, 46*(2), 394–409. https://doi.org/10.7202/003364ar

Li, T., Zhang, G., Do, D. Q., Yue, X., & Chen, W. (2025). Long-context LLMs struggle with long in-context learning. *Transactions on Machine Learning Research, 3*, 1–20. https://openreview.net/pdf/46ece7d97c904d3e89186a8f397075d8c663cd47.pdf

Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., & Liang, P. (2024). Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics, 12*, 157–173. https://doi.org/10.1162/tacl_a_00638

Lu, X., & Han, C. (2023). Automatic assessment of spoken-language interpreting based on machine-translation evaluation metrics: A multi-scenario exploratory study. *Interpreting, 25*(1), 109–143. https://doi.org/10.1075/intp.00076.lu

Macháček, D., Bojar, O., & Dabre, R. (2023). MT metrics correlate with human ratings of simultaneous speech translation. *Proceedings of IWSLT 2023*, 169–179. https://aclanthology.org/2023.iwslt-1.12.pdf

McCloskey, M., & Cohen, N. J. (1989). Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of Learning and Motivation, 24*, 109–165. https://doi.org/10.1016/S0079-7421(08)60536-8

NAATI. (2023). *Certified interpreter assessment rubrics*. https://www.naati.com.au/wp-content/uploads/2023/07/Certified-Provisional-Interpreter-Assessment-Rubrics.pdf

OpenAI. (2025, April 16). *ChatGPT (GPT-4o version)* [Large language model]. https://chat.openai.com/

Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002). BLEU: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 311–318. https://dl.acm.org/doi/10.3115/1073083.1073135

Pöchhacker, F. (2001). Quality assessment in conference and community interpreting. *Meta, 46*(2), 410–425. https://doi.org/10.7202/003847ar

Prandi, B. (2025). Computer-assisted interpreting (CAI) tools and CAI tool training. In E. Davitti, S. Braun, & T. Korybski (Eds.), *Routledge Handbook of Interpreting, Technology and AI* (pp. 123–144). Routledge. https://doi.org/10.4324/9781003053248

Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2023). Robust speech recognition via large-scale weak supervision. *Proceedings of the 40th International Conference on Machine Learning, Proceedings of Machine Learning Research, 202*, 28492–28518. https://proceedings.mlr.press/v202/radford23a.html

Reynolds, L., & McDonell, K. (2021). Prompt programming for large language models: Beyond the few-shot paradigm. *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, Article 314, 1–7. https://doi.org/10.1145/3411763.3451760

Rodriguez Gonzalez, E. (2024). *The use of automatic speech recognition in cloud-based remote simultaneous interpreting* [Doctoral dissertation, University of Surrey]. https://doi.org/10.15126/thesis.901147

https://doi.org/10.26034/cm.jostrans.2026.8425

Romero-Fresco, P. (2011). *Subtitling through speech recognition: Respeaking*. St. Jerome. https://doi.org/10.4324/9781003073147

Romero-Fresco, P., & Martínez, J. (2015). Accuracy rate in live subtitling: The NER model. In J. Díaz Cintas & R. Baños-Piñero (Eds.), *Audiovisual translation in a global context: Mapping an ever-changing landscape* (pp. 28–50). Palgrave Macmillan. https://link.springer.com/chapter/10.1057/9781137552891_3

Romero-Fresco, P., & Pöchhacker, F. (2017). Quality assessment in interlingual live subtitling: The NTR model. *Linguistica Antverpiensia, New Series: Themes in Translation Studies, 16*, 149–167. https://doi.org/10.52034/lanstts.v16i0.438

Sahoo, P., Singh, A. K., Saha, S., Jain, V., Mondal, S., & Chadha, A. (2024). A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv:2402.07927*. https://doi.org/10.48550/arXiv.2402.07927

Wang, B., Deng, X., & Sun, H. (2022). Iteratively prompt pre-trained language models for chain of thought. *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2714–2730. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.emnlp-main.174

Webson, A., & Pavlick, E. (2022). Do prompt-based models really understand the meaning of their prompts? In M. Carpuat, M.-C. de Marneffe, & I. V. Meza Ruiz (Eds.), *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 2300–2344). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.naacl-main.167

Wein, S., Te, I., Cherry, C., Juraska, J., Padfield, D., & Macherey, W. (2024). Barriers to effective evaluation of simultaneous interpretation. *Findings of the Association for Computational Linguistics: EACL 2024*, 209–219. https://doi.org/10.18653/v1/2024.findings-eacl.15

Zhao, J., Arthur, P., Haffari, G., Cohn, T., & Shareghi, E. (2021). It is not as good as you think! Evaluating simultaneous machine translation on interpretation data. *arXiv:2110.05213*. https://doi.org/10.48550/arXiv.2110.05213

Data availability statement

Appendices, working files and NTR evaluations used for comparing human to automated evaluations are available at: https://doi.org/10.17605/OSF.IO/EBRNP.

https://doi.org/10.26034/cm.jostrans.2026.8425