Developing a Web-Based Intelligent Language Assessment Platform Powered by Natural Language Processing Technologies

Sarah Löber[1,2], Björn Rudzewitz[1,2], Daniela Verratti Souto[1], Luisa Ribeiro-Flucht[1,2], Xiaobin Chen[1,2]

[1]Hector Research Institute of Education Sciences and Psychology, University of Tübingen, Germany
[2]LEAD Graduate School and Research Network, University of Tübingen, Germany
{sarah.loeber,bjoern.rudzewitz,luisa.ribeiro-flucht,
xiaobin.chen}@uni-tuebingen.de,
daniela.verratti-souto@student.uni-tuebingen.de

Abstract

We introduce ILAP, an intelligent language assessment platform and reusable module that streamlines the creation, administration and scoring of language proficiency tests supported by Natural Language Processing (NLP) technologies. As a first implementation, we realized an automatic pipeline for the Elicited Imitation Test (EIT), a popular test format that has been widely adopted in language learning research for general proficiency and formative assessments. The platform can be extended to other test formats and assessment types. ILAP is a valuable tool for standardizing data collection in Second Language Acquisition (SLA) and Intelligent Computer Assisted Language Learning (ICALL) research as well as serving as an application for classroom assessment. In this paper, we present the design of the system and a preliminary evaluation of Large Language Models (LLMs) for generating language errors for EIT items.

1 Introduction

Language assessment is a way for teachers and researchers to understand the current level of a learner's knowledge so that they can adjust their teaching or understand how language develops in the learner (Révész and Brunfaut, 2020; McNamara, 2000). Traditionally, language assessment has been done with tests of various formats, such as written tests with multiple choice, essay writing items, or spoken interviews. These tests are typically created manually, administered and graded by language teachers or researchers in school or lab settings, except for large-scale standardized tests such as the Test of English as a Foreign Language (TOEFL) or the International English Language Testing System (IELTS), which also include automatic forms of assessment (Evanini et al., 2015). The complexity of language assessment and the labor-intensiveness of language test creation, administration and grading are a major challenge for teachers and Second Language Acquisition (SLA) researchers, especially when the need to assess the students repeatedly and frequently arises. We therefore address these issues by creating a comprehensive language assessment system incorporating NLP. These technologies accelerate test implementation and scoring, making language testing feasible for a broader audience.

In the present paper, we demonstrate ILAP (Intelligent Language Assessment Platform), which is designed to facilitate the creation, administration, scoring, and reporting of results of language tests supported by technologies such as Automatic Speech Recognition (ASR), and generative AI technologies, in particular Text-to-speech (TTS) and Large Language Models (LLMs). The system features easy test creation with NLP leveraged item construction, convenient web-based test deployment, and automatic test response scoring and reporting. As a first instance, the system's implementation supports the Elicited Imitation Test (EIT) format, a popular test format that has been found to be effective in evaluating learners' general proficiency and to tap into their implicit language knowledge. An EIT targeting specific linguistic constructs can potentially also be used as a formative assessment tool to facilitate adaptive teaching.

In the following section, we will first justify the

Sarah Löber, Björn Rudzewitz, Daniela Verratti Souto, Luisa Ribeiro-Flucht and Xiaobin Chen. Developing a Web-Based Intelligent Language Assessment Platform Powered by Natural Language Processing Technologies. *Proceedings of the 13th Workshop on Natural Language Processing for Computer Assisted Language Learning (NLP4CALL 2024).* Linköping Electronic Conference Proceedings 211: 126–136.
126

choice of EIT as a valuable test format to be implemented in an intelligent language assessment system by reviewing the research behind the test format. We will then specify how ILAP supports the whole procedure of EIT-based assessment and the above-mentioned technologies used in the system. Furthermore, we provide a preliminary evaluation of our automatic scoring and the use of generative AI for generating ungrammatical test items. The paper concludes with an outlook of the project and future work.

1.1 The Elicited Imitation Test

In SLA research, numerous types of tests have been used to characterize learners' language proficiency, implicit or explicit knowledge of a language or their cognitive abilities. EIT, a popular test format among SLA researchers, is a sentence repetition task that requires the test taker to listen to the recordings of some sentences one at a time and then repeat the sentence they have just heard. Distractor questions (e.g. simple arithmetic calculations or judgement of the truthfulness of the sentence) are often asked between the audio playback and the repetition to prevent the test taker from relying on their phonological memory but rather require them to make use of their language system based on the meaning of the sentence. EITs have been used in a variety of ways, notably as a measure of implicit knowledge or general language proficiency (Ellis, 2005; Yan et al., 2016). Several studies corroborate the high validity of the test (Yan et al., 2016; Kostromitina and Plonsky, 2022), highlighting its efficacy as well as reliability. Furthermore, EITs show potential to serve as a placement test in language education (Yan et al., 2020) and as a teacher tool to assess second language (L2) learners' oral production skills in language classes (Campfield, 2017). Better still, research has found that it is an effective assessment format for various languages (Wu et al., 2023).

So far, the EIT has been administered in different formats, with different design implementations. For example, researchers have incorporated ungrammatical sentences (Erlam, 2006). Carefully created ungrammatical sentences are often used in EITs to test learners' specific grammatical knowledge (Spada et al., 2015). That is, whether a test taker can correct specific grammar errors in the repetition stage is an indicator of their implicit knowledge of the grammatical constructs. Scoring methods also vary: in some tests, items are scored on a binary basis, for instance, correct or incorrect for the use of the target structure only (Erlam, 2006), while others use a more fine-grained 5-point scale (Ortega et al., 2002) or even a percentage scale (Lonsdale and Christensen, 2011). Due to the different design implementations of EITs used in research, it is challenging to compare proficiency measures across studies. Therefore, there have been calls to enhance standardization of the tests (Isbell and Son, 2022; Kostromitina and Plonsky, 2022).

EIT items can also be designed to target specific grammar constructs that are the learning targets at different L2 developmental stages. For example, *third-person singular -s* or *mass/count nouns* are popular target constructs in previous L2 English studies (Kim and Godfroid, 2023). This makes the test an effective tool for formative assessment, but also poses a challenge to the test creator as they will need to not only find and write sentences with the target constructs, but also consider the sentence length, lexical frequency and other grammatical constructs in the sentence prompts. All of these factors have been found to affect the difficulty of test items as well as the validity of an EIT (Yan et al., 2016; Hendrickson et al., 2010). Users of EITs also face challenges from test administration and scoring, which traditionally requires the presence of a teacher or researcher in the classroom or lab to control the test procedure and to listen to the test responses for scoring. Hence, it is time-consuming, labor-intensive, and therefore difficult to scale.

We aim to address these issues by introducing ILAP, a web-based language assessment platform, where assessments of language proficiency can be created, administered and scored automatically. The first type of test integrated on the platform is an EIT pipeline.

1.2 Related work

Automating a language test requires automating several individual components involved in the testing process. While, to our knowledge, there is no fully automatic pipeline for the EIT developed yet that allows full flexibility, there have been studies on automating individual components of the test, such as item creation (Christensen et al., 2010) or scoring (Graham et al., 2008; Isbell et al., 2023). The findings of these studies show promising results for the feasibility of automated EITs.

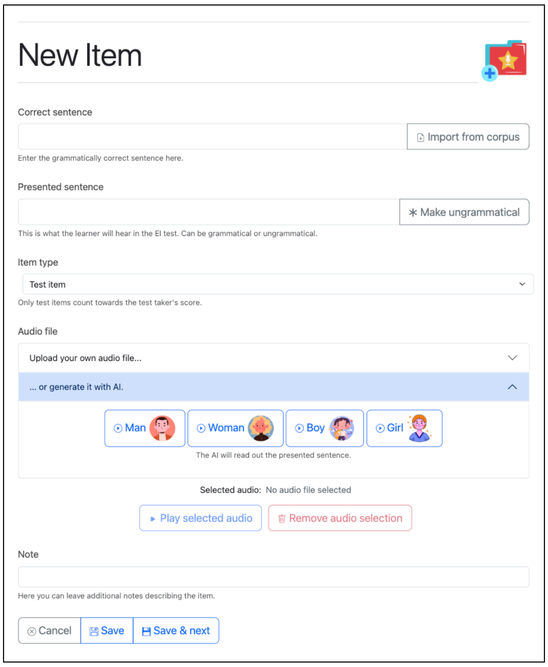
In the case of automatic scoring, studies proposing solutions have focused on transcribing test takers' responses with automatic speech recognition (ASR) and implementing rules for scoring the transcriptions. For example, utilizing ASR and transcription scoring metrics based on string edit distance, Isbell et al. (2023) were able to achieve high correlations with human scoring ($r > .90$) across all items on the Korean EIT. Likewise, Graham et al. (2008) reported high correlations for a method using ASR and binary scoring on the syllable level.

Pertaining to the further automation of EITs, Christensen et al. (2010) utilized a language corpus for the automatic and flexible selection of elicited imitation test items with their item selection tool. The automatically selected EIT showed higher correlations with the speaking language achievement test (SLAT) than previous EITs.

The EIT is often administered in a lab, as part of data collection for studies. An alternative would be to administer the test online, allowing for more flexibility, easier processing of the responses and potentially reaching more participants. Some studies have administered the EIT in this way, with web-based and lab-based EITs showing no significant difference in their validity (Kim et al., 2024). However, Kim et al. (2024) found weaker correlations, albeit non-significantly, for the web-based EIT and TOEFL scores than for the lab-based EIT and TOEFL scores when taking only ungrammatical items into account. According to the authors, this could result from a lack of immediate feedback in the web-based EIT. Informed by and building on previous efforts to automate EIT creation, administration, and scoring, we implement the process in a newly developed intelligent language assessment system that utilizes latest AI technologies. The next section provides more details.

2 System overview

ILAP is a web-based application that is mobile-friendly and compatible with most devices. The back end is coded in Java, while the web front end utilizes JavaScript and the Bootstrap framework. There are two interfaces offered: a test creator interface as well as a test taker view, both of which require user profiles and accounts with different roles. In the following, we describe the test creation, administration and scoring procedure with ILAP.



New Item

Correct sentence

Enter the grammatically correct sentence here.

Presented sentence

This is what the learner will hear in the EI test. Can be grammatical or ungrammatical.

Item type

Test item

Only test items count towards the test taker's score.

Audio file

Upload your own audio file...

... or generate it with AI.

The AI will read out the presented sentence.

Selected audio: No audio file selected

Note

Here you can leave additional notes describing the item.

Figure 1: Interface for creating new test items

2.1 Test creation

Test creators start by creating a test collection. This automatically generates a unique and random 4-character access code that the test creator can give to the participants to take the test. In the next step, tests can be added to the test collection. The choice for letting the user add different tests into the test collection was made with the future integration of new test types in mind. This will allow the integration of several separate test components into one test collection, e.g. an EIT followed by a reading comprehension test. When adding a new test, users can specify the name, description and visibility of the test. Tests with visibility set to public can be shared among test creators. Afterwards, users can add the test type. In the first step, we implemented an elicited imitation test type. Within the created test, users can then manage instructions, items and settings or preview their test. Figures 5, 6, and 7 in Appendix A.1 show the test creation process.

Instructions The instruction interface allows the user to add instructions, including their title and text. Furthermore, users specify at what point during the test an instruction is shown, e.g. before practice items or before each item. Test creators can add any number of instructions for the test, with each instruction appearing on a separate page in the test-taker interface. Figure 8 in Appendix A.2 shows a screenshot of this interface.

Items Figure 1 shows the item interface from the test creator perspective, which supports adding grammatical as well as ungrammatical items. Test creators can add their own sentences or choose sentences from the provided sentence corpus. For the latter case, we annotated about 95.000 extracted sentences from the Spotlight corpus (Weiss et al., 2021) with constructs from the English Grammar Profile (EGP, O'Keeffe and Mark, 2017) using an in-house EGP annotator. Users can search, select and import sentences from the corpus, filter by grammatical construct, and also edit the sentences for their items.

The interface supports both a manual and an automatic creation of ungrammatical variants of sentences. We implemented a component incorporating `GPT-4o` through the OpenAI API (OpenAI, 2024a) to automatically produce ungrammatical variants of the sentence, i.e. simulating the output of mal-rules (e.g. Sleeman, 1985) on the correct sentence. The generated ungrammatical sentence is based on the user input. Users can enter the correct sentence and select the "Make ungrammatical" button, upon which the generated ungrammatical sentence is displayed in the "Sentence" field in the interface. A more elaborate evaluation on our choice for using `GPT-4o` for this functionality, including quantitative and qualitative human assessments on the error generation, is provided in Section 3.

Furthermore, an item can be classified as practice or test item. Audio files for items can either be uploaded or automatically generated. For this functionality, we are using the text-to-speech service from Amazon Web Services (AWS) [1]. Lastly, a note can be added to describe an item.

Settings Test creators can control all settings related to a test by overriding the default settings of tests with their own values. For example, they can control the duration of the recording of responses by test takers, whether belief statement checks are shown after items are shown, and more. This interface is displayed in Figure 9 in Appendix A.3.

2.2 Test administration

Each test collection is created with the status "in editing". As long as a test has this status, it cannot be started by test takers. In case an access code for a test collection that is not released is entered, test takers get a warning message informing them that the test is not available yet. Test creators can control the release of a test by updating its status to "released". At this point, the test becomes available and cannot be edited anymore in order to avoid problems related to test taker data referring to an outdated version of the test, ensuring a valid data collection process.

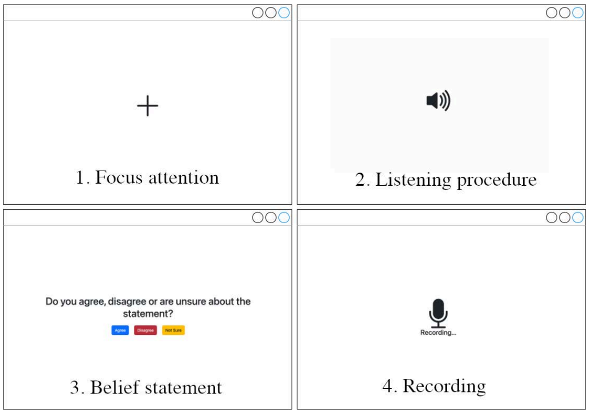


Figure 2: Test taker perspective of item procedure

Test takers can access the released test via the test taker interface by entering the provided access code. The item procedure, using the default settings, is shown in Figure 2.

2.3 Scoring and results

To allow for full flexibility for the test creator, completed EITs can be scored manually as well as automatically. When a test has been taken by the user, test creators can access the result overview via the corresponding test in the "My tests" interface. In this overview, test results are grouped by the progress of test takers. Tests which have been started, but not yet finished by the user are also shown. Responses for finished tests can either be scored automatically (all at once or item by item) or manually in the performance overview, where the test takers' audio transcriptions and the string edit distance measures to the correct sentence, converted to a percentage, are displayed.

For the automatic scoring algorithm, following the work of Isbell et al. (2023), we use the transcription of the test response and string edit distance measures to calculate the test score. The recorded audio files of the test takers are automatically transcribed and the transcription is compared to the correct sentence for each item specified by

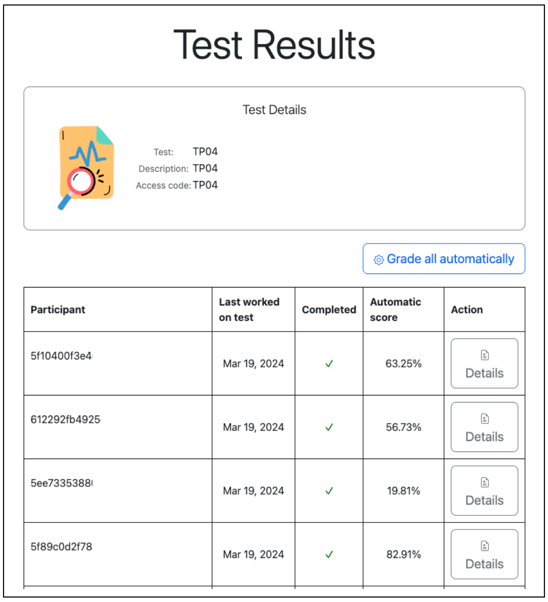
[1] `https://aws.amazon.com/`

the test creator. For transcription of test-taker responses, we are using the Whisper-large-v2 model through the OpenAI transcriptions API (OpenAI, 2024b). Our choice of Whisper for response transcription is based on previous research (Bear et al., 2023) showing that Whisper has the lowest word error rate (WER) when compared to other commercial ASR providers on ungrammatical and grammatical sentences from L2 speakers.

For string edit distance comparison, the system first normalizes the transcription string as well as the target string by converting the characters to lowercase and removing the non-word characters as well as whitespace characters. We decided on this process of normalization after noticing that the ASR would occasionally add punctuation characters, for example adding a question mark when raising the voice at the end of a sentence. After this process, the mean of three string distance measures is computed: Levenshtein, Jaro-Winkler and Jaccard distance. We are using the mean of these measures in order to retain the different measure characteristics while also making the result more accessible by offering only one score to the test creator. For making these scores more intuitive, the mean of the three measures is converted to a percentage on the item-level as well as the test-level, ranging from 0 to 100. Figure 3 shows the scoring interface on the test-level. The system offers an additional field on the item-level for a manual score in case test creators want to apply their own scoring metric, e.g. a school grading system. A screenshot of the scoring interface on the item-level can be found in Figure 10 in Appendix A.4.

2.4 Preliminary testing of scoring functionality

For preliminary testing of our scoring implementation, we manually scored 22 EITs taken with our system. Scoring each item on a scale of 0-4, we followed the established scoring scheme of Ortega et al. (2002), with the sum of all item scores as the total EIT score. The EITs consisted of 24 items, resulting in a maximum total score of 96 for the manual scoring. We then correlated the total manual EIT scores with the total automatic scores of these 22 tests in ILAP. We achieved a correlation of $r = .95$ across items, which is in line with previous studies employing this approach (Isbell et al., 2023). Figure 4 shows the correlation of manual and automatic scores. The x-axis shows the manual scores, ranging from 0-96 points and the y-axis shows the similarity score of the target answer and the learner answer in percentages. The line shows the fitted linear model with 95% confidence interval.



Test Results

Test Details

Test: TP04
Description: TP04
Access code: TP04

Participant	Last worked on test	Completed	Automatic score	Action
5f10400f3e4	Mar 19, 2024	✓	63.25%	Details
612292fb4925	Mar 19, 2024	✓	56.73%	Details
5ee7335388	Mar 19, 2024	✓	19.81%	Details
5f89c0d2f78	Mar 19, 2024	✓	82.91%	Details

Figure 3: Interface for scoring responses on the test level

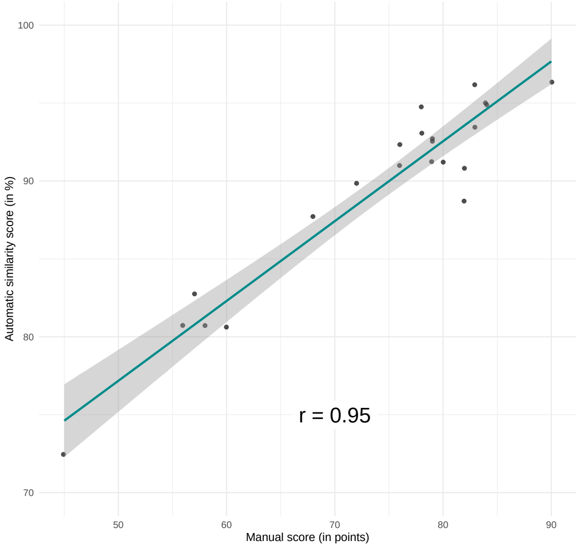


Figure 4: Correlation of manual EIT scores and automatic similarity scores. The grey area represents the 95% confidence interval of the fitted linear model.

3 Preliminary evaluation of LLMs for ungrammatical item generation

In order to evaluate whether the changes introduced by LLMs can be considered realistic errors, i.e. errors that are plausible to expect from learners, we conducted a preliminary evaluation for our ungrammatical sentence generation functionality in ILAP. We compared GPT-3.5-turbo, GPT-4, GPT-4o and Claude 3 Haiku on their performance of the generation of ungrammatical sentences. We used sixteen grammatical versions of existing EIT items from previous tests (Erlam, 2006; Spada et al., 2015; Godfroid and Kim, 2021) and prompted the models by providing examples from previously used ungrammatical EIT items and specifying the limit for the amount of changes to be made in the sentence. Our evaluation focused on quantitative as well as qualitative aspects.

3.1 Quantitative evaluation

For the quantitative evaluation of the plausibility of errors, there were no error-annotated learner corpora available containing specifically the test items we used. Therefore, we conducted our evaluation with the output of a mal-rule-based, generative approach based on actual learner error patterns. Mal-rules are patterns to parse or generate learner language that model specific misconceptions or errors (Sleeman, 1985).

An example of extensive mal-rule usage is the successful FeedBook system (Meurers et al., 2019), which is an ICALL system for English as a second language that incorporates an automatic feedback generation approach capable of generating a wide range of possible errors based on a well-formed target answer (Rudzewitz et al., 2018). The feedback generation component works by iteratively applying mal-rules derived from a corpus of actual learner errors to an input string and thereby automatically generating a wide range of ill-formed variants of input string along with error diagnoses (Ziai et al., 2018). Those variants can then be aligned with answers produced by learners, and if there is a match, the diagnosis associated with a generated variant is used to display a scaffolding feedback message to the learner.

Since the mal-rules included in FeedBook represent generalizations of actually observed learner errors, we employed the overlap between the output of the FeedBook feedback generation and the output of the LLMs as a criterion to assess the plausibility of the errors generated by the LLMs. To this end, we let the FeedBook feedback generation component generate all possible variants based on ten experimental test items, and computed the degree of overlap between the sentences from this approach with the sentences generated by the LLMs. Table 1 shows the results.

Model	FeedBook Overlap
GPT-3.5-turbo	27.3
GPT-4	27.3
GPT-4o	81.8
Claude 3 Haiku	63.6

Table 1: Overlap (in percentages) between the output of different LLMs with the output of the FeedBook mal-rule-based generative approach

Since not all constructs in all sentences were covered by the ICALL system's generative approach due to the fact that the FeedBook was designed for a specific grade, we restricted the comparison to those sentences where the FeedBook generated alternative variants, which were ten out of sixteen experimental test items. An example of generated errors by the four LLMs and FeedBook can be found in Table 2.

The results show that GPT-4o produced the highest overlap with the output from the mal-rule generation approach.

3.2 Human evaluation

We also conducted human evaluation with the sentences generated by the four models. We asked human raters to evaluate the ungrammatical sentences on 5-point Likert scales on three different dimensions, namely

- *Naturalness of Error (NoE)*: this sentence contains an error that is characteristic of an error produced by language learners
- *Retention of meaning*: this sentence retains the meaning of the correct sentence
- *Adherence to prompt*: the output adheres to the prompt given to the LLM

Seven human evaluators, all experts in linguistics with teaching experience, rated the same 16 sentences generated by each model without knowing which model the sentences were from, resulting in a total of 64 sentences per evaluator. Evaluators indicated their agreement to the dimensions above on a 5-point Likert scale ranging from 1 - Strongly disagree to 5 - Strongly agree. Table 3 shows the results of the human evaluation [2].

[2] The data and analysis scripts are available under https://osf.io/tjn4v/

Model	Generated error sentence(s)
GPT-3.5-turbo	Family names is often changed after marriage.
GPT-4	Family names are often changed after marriage it.
GPT-4o	Family name are often changed after marriage.
Claude 3 Haiku	Family names are often change after marriage.
FeedBook	Family names are often change after marriage. Family names often changed after marriage. Family names often changes after marriage. Family names is often changed after marriage. Family names be often changed after marriage. . . .

Table 2: Examples of LLM-generated errors and FeedBook generated error variants on the input sentence "Family names are often changed after marriage.". Input sentence taken from Spada et al. (2015).

We performed additional statistical analyses on the evaluation data in R. Given the small sample, we conducted Shapiro-Wilk tests to assess the normality of the distribution for each dimension. The Shapiro-Wilk tests showed significant deviations from normality, confirming our data were not normally distributed. Therefore, we opted for non-parametric tests for further analysis. A Kruskall-Wallis test showed a significant difference of means on Naturalness of errors ($H(3) = 14, p = 0.002$) and Retention ($H(3) = 10, p = 0.02$). Pairwise Mann-Whitney U comparisons with Bonferroni corrections were conducted to determine which specific models differed. The results revealed that GPT-4o significantly outperformed GPT-3.5-turbo on Naturalness of errors ($p = 0.01$). Furthermore, GPT-4o significantly outperformed Claude 3 Haiku on Retention ($p = 0.03$).

Model	NoE	Retention	Adherence
GPT-3.5-turbo	3.41	4.42	4.55
GPT-4	3.79	**4.71**	4.79
GPT-4o	**4.09**	4.66	**4.83**
Claude 3 Haiku	3.88	4.38	4.71

Table 3: Mean ratings of LLM generated ungrammatical sentences on three dimensions.

3.3 Results and discussion

Based on the results of the preliminary evaluation, we decided to use GPT-4o for generating ungrammatical variants of sentences in our system. Our quantitative evaluation shows the high overlap between GPT-4o output and the mal-rule-based approach, suggesting that GPT-4o generates plausible learner errors. The human evaluation strengthened this finding, with GPT-4o achieving the highest ratings in naturalness of errors as well as adherence (although not significantly). GPT-4 also seemed to perform well in the human evaluation, achieving the highest ratings in retention of meaning. However, the quantitative evaluation showed a low overlap between the mal-rule-based generative approach and the sentences generated by GPT-4, which might be due to the ICALL system's limited scope in producing more advanced learner errors, since it only covers specific grammatical constructs. For example, the sentence "Birthday cards have been emailed since hundreds of years.", with the same error being generated by both GPT-4 and GPT-3.5-turbo, had no matching variant in the FeedBook output, but was rated plausibly by humans. This is possibly due to the since/for construct not being included in the ICALL system.

It is also noteworthy that out of all dimensions, all models score lowest on the NoE dimension, meaning that the errors generated by the models were not rated as highly natural or being highly characteristic of language learners by the human evaluators. This observation could indicate that commercial LLMs might not excel at generating mal-formed language, but rather have been demonstrated to be highly effective for grammatical error correction (e.g. Katinskaia and Yangarber (2024)). Better results for error generation could be achieved with a model fine-tuned for this specific task (Bryant et al., 2023).

4 Limitations

There are some limitations to our system. First, the small amount of validation data demands cautiousness when making any claims about the effectiveness of ILAP or our scoring functionality. For this reason, we are striving for wider deployment of the platform to collect data from different and larger groups, for example in schools or learning environments. Secondly, questions about the effect of the use of technology in the creation of the EIT items remain open. In the future, we plan to investigate to what extent technology can be used in creating language proficiency test items and the effect on test validity. Thirdly, there currently is a lack of test types in ILAP. We are working on implementing more test types, which would make the platform more versatile and adaptable to more use cases.

Additionally, the system could provide even more support in the item creation process, for instance an assessment of an item's difficulty. This functionality is currently not integrated into the platform. Arguably, this would make it easier to create EIT items of varying difficulty. Future research could focus on the automatic item difficulty prediction of EIT items, where important progress has been made in the context of computer-adaptive testing (Settles et al., 2020).

As discussed in Section 3.3, we conducted a novel but preliminary evaluation for assessing the output of LLMs for error generation in both a quantitative and qualitative way. Our quantitative approach for evaluation is arguably not without flaws and might benefit from including a larger set of data as well as more diverse resources and approaches, such as an error-annotated corpus, in order to evaluate the generation of ungrammatical sentences. This would also enable further research to go beyond the scope of EIT items.

5 Conclusion and future work

We presented a system for automatic language assessment as well as data collection and computer-assisted scoring. We included a pipeline for elicited imitation tests, which can be used for both research and education. Furthermore, we described our preliminary evaluation of the integrated scoring functionality and presented an approach for the evaluation of LLMs for generating ungrammatical sentences. To the best of our knowledge, this type of evaluation has not been performed before. With the integration of the EIT we have made an important first step in enabling automatic language assessment and standardizing proficiency tests in SLA research useful for teachers, researchers and test creators. The benefit of such a system can be of importance to other domains, such as ICALL. As Ruiz et al. (2023) stated, not all ICALL systems currently offer a built-in functionality for collecting test results for SLA research, leading the authors to emphasize the need for reusable modules. Since ILAP can potentially be integrated into other systems, it can be used to simplify the process of testing for ICALL systems.

In the future, we will expand the platform with a teacher dashboard view and implement more test types to make the system more relevant for usage in schools. Currently, we have started the deployment of the system for studies, including a study to test the effects of automatic speech synthesis on test validity and other studies on factors (e.g., speech rate) that might affect test performance and scoring, and, consequently, the reliability of EITs. As for our ungrammatical item generation analysis, we plan to build on and extend this analysis by increasing the sample size for the analysis to cover more types of learner errors. Furthermore, we intend to error-annotate the output of the LLMs according to annotation criteria for learner corpora in order to be able to compare the frequency of the generated error types with the frequency of the error type in learner corpora and use this information as an additional criterion for the plausibility of the errors and quality of the LLM output. Additionally, we plan to explore the effects of fine-tuning a large language model for this task specifically based on error-annotated learner corpora.

The ILAP system is currently available at https://ilap.kibi.group.

Acknowledgements

We would like to thank the two anonymous NLP4CALL reviewers for their helpful feedback. This project is funded by the German Ministry of Education and Science (BMBF) under the funding number 01IS22076.

References

Elizabeth Bear, Stephen Bodnar, and Xiaobin Chen. 2023. Learner and linguistic factors in commercial ASR use for spoken language practice: A focus on form. In *Proc. 9th Workshop on Speech and*

Language Technology in Education (SLaTE), pages 161–165.

Christopher Bryant, Zheng Yuan, Muhammad Reza Qorib, Hannan Cao, Hwee Tou Ng, and Ted Briscoe. 2023. Grammatical error correction: A survey of the state of the art. *Computational Linguistics*, 49(3):643–701.

Dorota E. Campfield. 2017. Lexical difficulty – using elicited imitation to study child L2. *Language Testing*, 34(2):197–221.

Carl Christensen, Ross Hendrickson, and Deryle Lonsdale. 2010. Principled construction of elicited imitation tests. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Rod Ellis. 2005. Measuring implicit and explicit knowledge of a second language: A psychometric study. *Studies in Second Language Acquisition*, 27(2):141–172.

Rosemary Erlam. 2006. Elicited imitation as a measure of L2 implicit knowledge: An empirical validation study. *Applied Linguistics*, 27(3):464–491.

Keelan Evanini, Michael Heilman, Xinhao Wang, and Daniel Blanchard. 2015. Automated scoring for the TOEFL Junior® comprehensive writing and speaking test. *ETS Research Report Series*, 2015(1):1–11.

Aline Godfroid and Kathy Minhye Kim. 2021. The contribution of implicit-statistical learning aptitude to implicit second-language knowledge. *Studies in Second Language Acquisition*, 43(3):606—-634.

C. Ray Graham, Deryle Lonsdale, Casey Kennington, Aaron Johnson, and Jeremiah McGhee. 2008. Elicited imitation as an oral proficiency measure with ASR scoring. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Ross Hendrickson, Meghan Aitken, Jeremiah McGhee, and Aaron Johnson. 2010. What makes an item difficult? A syntactic, lexical, and morphological study of elicited imitation test items. In *Selected Proceedings of the 2008 Second Language Research Forum*, pages 48–56. Cascadilla Proceedings Project Somerville, MA.

Daniel R. Isbell, Kathy Minhye Kim, and Xiaobin Chen. 2023. Exploring the potential of automated speech recognition for scoring the Korean Elicited Imitation Test. *Research Methods in Applied Linguistics*, 2(3):100076.

Daniel R. Isbell and Young-A Son. 2022. Measurement properties of a standardized elicited imitation test: An integrative data analysis. *Studies in Second Language Acquisition*, 44(3):859–885.

Anisia Katinskaia and Roman Yangarber. 2024. GPT-3.5 for grammatical error correction. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7831–7843, Torino, Italia. ELRA and ICCL.

Kathy Minhye Kim and Aline Godfroid. 2023. The interface of explicit and implicit second-language knowledge: A longitudinal study. *Bilingualism: Language and Cognition*, 26(4):709–723.

Kathy Minhye Kim, Xiaoyi Liu, Daniel R Isbell, and Xiaobin Chen. 2024. A comparison of lab-and web-based elicited imitation: Insights from explicit-implicit L2 grammar knowledge and L2 proficiency. *Studies in Second Language Acquisition*, pages 1–22.

Maria Kostromitina and Luke Plonsky. 2022. Elicited imitation tasks as a measure of L2 proficiency: a meta-analysis. *Studies in Second Language Acquisition*, 44(3):886–911.

Deryle Lonsdale and Carl Christensen. 2011. Automating the scoring of elicited imitation tests. In *Proc. Machine Learning in Speech and Language Processing (MLSLP 2011)*, pages 16–20.

Timothy Francis McNamara. 2000. *Language Testing*. Oxford University Press.

Detmar Meurers, Kordula De Kuthy, Florian Nuxoll, Björn Rudzewitz, and Ramon Ziai. 2019. Scaling up intervention studies to investigate real-life foreign language learning in school. *Annual Review of Applied Linguistics*, 39:161–188.

OpenAI. 2024a. OpenAI API. Software.

OpenAI. 2024b. OpenAI API Whisper. Software.

Lourdes Ortega, Noriko Iwashita, John M Norris, and Sara Rabie. 2002. An investigation of elicited imitation tasks in crosslinguistic SLA research. In *Second Language Research Forum, Toronto*, pages 3–6. Paper presentation.

Anne O'Keeffe and Geraldine Mark. 2017. The English Grammar Profile of learner competence: Methodology and key findings. *International Journal of Corpus Linguistics*, 22(4):457–489.

Andrea Révész and Tineke Brunfaut. 2020. Validating assessments for research purposes. In *The Routledge Handbook of Second Language Acquisition and Language Testing*, pages 21–32. Routledge.

Björn Rudzewitz, Ramon Ziai, Kordula De Kuthy, Verena Möller, Florian Nuxoll, and Detmar Meurers. 2018. Generating feedback for English foreign language exercises. In *Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications*, pages 127–136.

Simón Ruiz, Patrick Rebuschat, and Detmar Meurers. 2023. Supporting individualized practice through intelligent CALL. In Yuichi Suzuki, editor, *Practice and Automatization in Second Language Research*, 1 edition, pages 119–143. Routledge, New York, NY.

Burr Settles, Geoffrey T. LaFlair, and Masato Hagiwara. 2020. Machine Learning–Driven Language Assessment. *Transactions of the Association for Computational Linguistics*, 8:247–263.

Derek Sleeman. 1985. Inferring (mal) rules from pupils' protocols. In *Selected and updated papers from the proceedings of the 1982 European conference on Progress in artificial intelligence*, pages 30–39.

Nina Spada, Julie Li-Ju Shiu, and Yasuyo Tomita. 2015. Validating an elicited imitation task as a measure of implicit knowledge: Comparisons with other validation studies. *Language Learning*, 65(3):723–751.

Zarah Weiss, Xiaobin Chen, and Detmar Meurers. 2021. Using broad linguistic complexity modeling for cross-lingual readability assessment. In *Proceedings of the 10th Workshop on NLP for Computer Assisted Language Learning*, pages 38–54.

Shu-Ling Wu, Yee Pin Tio, and Yuening Zhao. 2023. Examining the comparability of parallel English and Chinese elicited imitation tasks. *Research Methods in Applied Linguistics*, 2(3):100058.

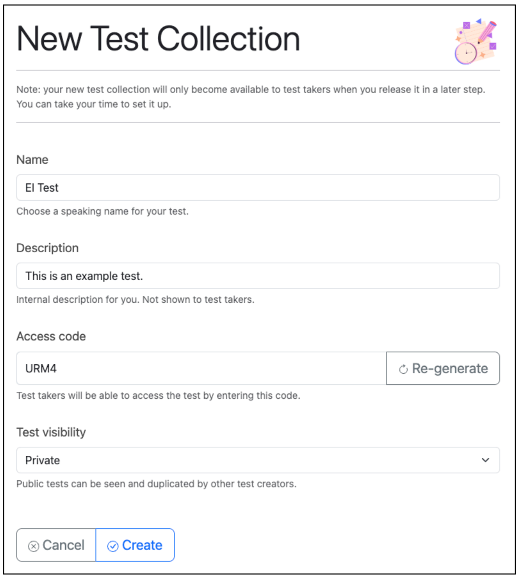
Xun Yan, Yuyun Lei, and Chilin Shih. 2020. A corpus-driven, curriculum-based chinese elicited imitation test in US universities. *Foreign Language Annals*, 53(4):704–732.

Xun Yan, Yukiko Maeda, and April Ginther. 2016. Elicited imitation as a measure of second language proficiency: A narrative review and meta-analysis. *Language Testing*, 33(4):497–528.

Ramon Ziai, Björn Rudzewitz, Kordula De Kuthy, Florian Nuxoll, and Detmar Meurers. 2018. Feedback strategies for form and meaning in a real-life language tutoring system. In *Proceedings of the 7th workshop on NLP for Computer Assisted Language Learning*, pages 91–98.

Appendix A

A.1 Test creation process



New Test Collection

Note: your new test collection will only become available to test takers when you release it in a later step. You can take your time to set it up.

Name

EI Test

Choose a speaking name for your test.

Description

This is an example test.

Internal description for you. Not shown to test takers.

Access code

URM4

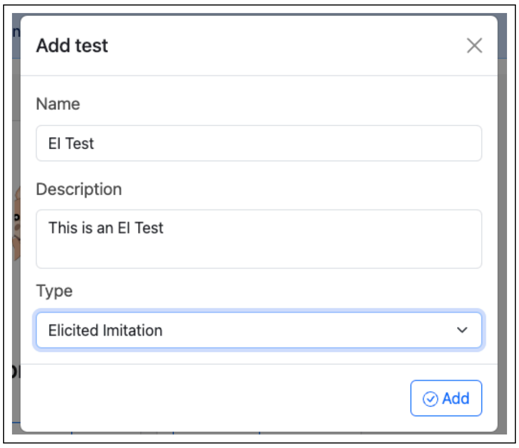
Test takers will be able to access the test by entering this code.

Test visibility

Private

Public tests can be seen and duplicated by other test creators.

Figure 5: User interface to create a new test



Add test

Name

EI Test

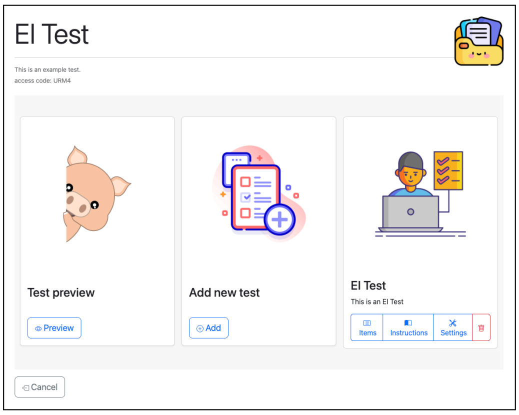
Description

This is an EI Test

Type

Elicited Imitation

Figure 6: Add test page with the test type Elicited Imitation



EI Test

This is an example test.
access code: URM4

Test preview

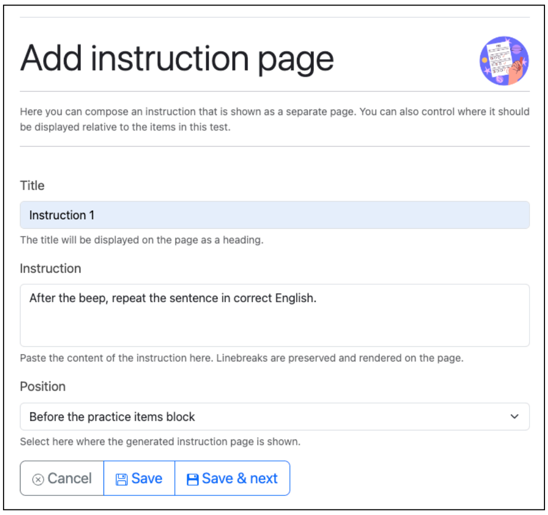
Add new test

EI Test

This is an EI Test

Figure 7: Test component management, where instructions, items and settings can be edited and added

A.2 Instructions



Add instruction page

Here you can compose an instruction that is shown as a separate page. You can also control where it should be displayed relative to the items in this test.

Title

Instruction 1

The title will be displayed on the page as a heading.

Instruction

After the beep, repeat the sentence in correct English.

Paste the content of the instruction here. Linebreaks are preserved and rendered on the page.

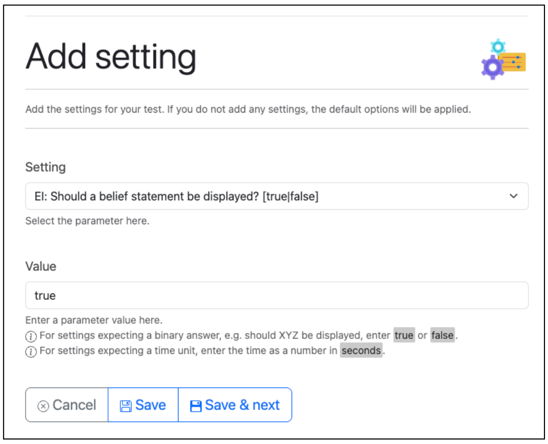
Position

Before the practice items block

Select here where the generated instruction page is shown.

Figure 8: Interface for adding instruction pages for tests

A.3 Settings



Add setting

Add the settings for your test. If you do not add any settings, the default options will be applied.

Setting

EI: Should a belief statement be displayed? [true|false]

Select the parameter here.

Value

true

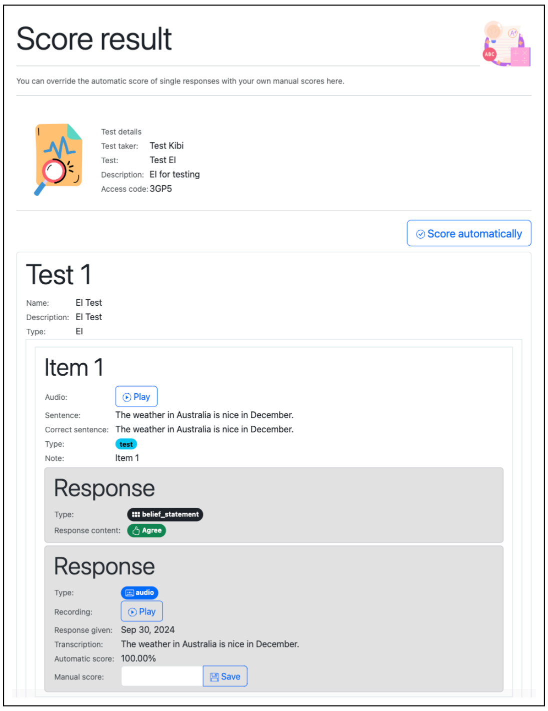
Enter a parameter value here.

ⓘ For settings expecting a binary answer, e.g. should XYZ be displayed, enter true or false.

ⓘ For settings expecting a time unit, enter the time as a number in seconds.

Figure 9: Interface for adding test settings

A.4 Item-level scoring



Score result

You can override the automatic score of single responses with your own manual scores here.

Test details

Test taker: Test Kibi

Test: Test EI

Description: EI for testing

Access code: 3GP5

Test 1

Name: EI Test

Description: EI Test

Type: EI

Item 1

Audio: Play

Sentence: The weather in Australia is nice in December.

Correct sentence: The weather in Australia is nice in December.

Type: test

Note: Item 1

Response

Type: belief_statement

Response content: Agree

Response

Type: audio

Recording: Play

Response given: Sep 30, 2024

Transcription: The weather in Australia is nice in December.

Automatic score: 100.00%

Manual score:

Figure 10: Interface to score responses to individual items