Alma Mater Studiorum Università di Bologna

DIPARTIMENTO DI INTERPRETAZIONE E TRADUZIONE

Corso di Laurea magistrale Specialized Translation (classe LM - 94)

TESI DI LAUREA

in Corpora, Linguistics and Technology Lab

Learner Corpora and Artificial Intelligence: Towards Error Annotation of a Corpus of Italian EFL Students' Interactions with Chatbots

CANDIDATA:

Arianna Paradisi

RELATORE:

Adriano Ferraresi

CORRELATRICE

Maja Miličević Petrović

Anno Accademico 2023/2024
Terzo Appello

Table of Contents

List of Figures

List of Tables

Abstract

This thesis is part of the *UNITE — Universally inclusive technologies to practice English* project, which aims to create and analyse a learner corpus based on interactions between Italian students of English as a Foreign Language (EFL) and chatbots. The thesis specifically presents two case studies, one on error annotation of a sample of texts from the corpus, and another on the possibility of using ChatGPT for automating the error annotation process. The first case study involved the error annotation of students' conversational turns from 23 texts using the Louvain Error Tagging Manual Version 2.0, which resulted in the refinement of the error taxonomy so that it could align with the conversational nature of the UNITE corpus. Among other results, the distribution of errors annotated using the refined error tagset showed that the corpus presents several features commonly associated with digitally-mediated-communication, with orthographic and morphological errors being the most frequent type of linguistic errors. The second case study consisted of a proof-of-concept experiment where a custom GPT powered by the ChatGPT-4o model was created and used for error annotating four texts from the sample manually annotated corpus. By comparing the GPT's output with human annotations, results on accuracy revealed that the chatbot was able to reach an acceptable level of accuracy. This means that, even if with due attention, it may be used as a preliminary instrument for error annotation, followed by an accurate revision and post-editing.

Keywords: artificial intelligence, large language models, chatbots, language learning, dialogue-based Computer-Assisted Language Learning, learner corpora, corpus annotation, error annotation

Acknowledgments

I would like to express my deepest gratitude to everyone who supported me through this important journey. First, I'm extremely grateful to my thesis supervisor, Professor Adriano Ferraresi, and co-supervisor, Professor Maja Miličević Petrović, for giving me the opportunity to contribute to the UNITE project, for their trust, and for their exceptional guide and constant support.

I would also like to extend my heartfelt thanks to Daniele Polizzi and Giada Palmieri, UNITE's research fellow, for being invaluable companions on this journey. The opportunity to work together, share idea, and support one other was an essential part for the outcome of this work.

My appreciation also goes to all the members of the Corpora, Linguistics, Technology group for providing me with the opportunity to share opinions on some of the steps of my research with you.

Finally, a very special thanks goes to Professor Magali Paquot and all members of the Centre for English Corpus Linguistics (Université catholique de Louvain) for welcoming me to Louvain and for their valuable insights and advice on my project.

1. Introduction

During the last three years, the concepts of ‘artificial intelligence (AI)’ and ‘chatbot’ have become part of our lives more than ever before. The release of ChatGPT in November 2022 has completely changed our way of looking at and perceiving this kind of technology, which is becoming more and more sophisticated. From text summarisation to coding skills, AI-powered chatbots’ ability to generate very accurate outputs has led to exploring their application in many different fields, including language teaching and language learning. Can these tools be effectively used as conversational partners to practice a foreign language? One of the main problems that many language students face is that, unless they have the opportunity to travel, they may lack possibilities of interacting with native speakers and practicing their language skills outside the classroom. For this reason, chatbots may be an excellent instrument to help students improve their proficiency. However, to do this, research is needed to evaluate the feasibility and effectiveness of integrating these tools in educational contexts. Until the present day, studies have mainly focused on learners’ motivation and satisfaction in the use of chatbots, but a research gap is still present when it comes to the analysis of their actual interactions.

This thesis is conducted within the *UNITE — Universally inclusive technologies to practice English* project[1], which aims to fill this research gap by creating and analysing a learner corpus based on interactions between students and chatbots, with the ultimate goal of supporting an efficient integration of AI technology in language education. The specific focus of this thesis is to contribute to the error annotation of the UNITE corpus—a significant step for future analysis of both learner productions and chatbot’s reactions to students’ errors. The two case studies presented aimed to: (1) identify the error taxonomy that best suits the conversational nature of the UNITE corpus, (2) investigate the possibility of automating the error annotation process through the use of AI-powered chatbots like ChatGPT. Both studies should help UNITE’s researchers to streamline the error annotation step of the project.

The thesis is structured in five main parts. Chapter 2 provides a theoretical background of the concepts useful for delimiting the field of which the project is part. Chapter 3 describes the UNITE corpus, including data collection, as well as metadata and structural annotations. It also presents the Louvain Error Tagging Manual (Granger *et al.*, 2022) and the UCLouvain Error Editor (UCLEE) (Granger *et al.*, 2023). Chapter 4 focuses on the first case study conducted, which consisted of using both the Louvain manual and error editor to annotate errors in

[1] https://site.unibo.it/unite/en

students' conversational turns in a sample of texts from the UNITE corpus, leading to the refinement of the error taxonomy. Chapter 5 presents the second case study—a proof-of-concept experiment to test ChatGPT's accuracy in error annotation tasks. Finally, Chapter 6 discusses the results obtained in both studies and presents heir limitations, as well as suggestions for future research.

2. Background

2.1. Overview

This chapter provides the theoretical background on the topics relevant to the project carried out in this thesis, starting with Section 2.2 that defines the concept of 'chatbots' and related technologies, such as machine learning and natural language processing. It continues by tracing the evolution of chatbots from early systems to state-of-the-art models, and by describing their main applications, including in educational settings. The pedagogical role of chatbots is then further outlined in Section 2.3, followed by the introduction of the concept of Computer-Assisted Language Learning (CALL) and the practical applications of chatbots in language teaching and learning. Subsequent sections focus on another important aspect of this thesis—corpus linguistics and language learning. Section 2.4 introduces the role of Learner Corpus Research, illustrates the design and uses of learner corpora, and discusses annotation practices, with a specific focus on error annotation. The integration of chatbots for corpus annotation tasks is then discussed in Section 2.5. Finally, Section 2.6 introduces the UNITE project—which combines the aforementioned topics by investigating a learner corpus based on interactions between language learners and chatbots—by providing an overview of the project's scope and its main objectives.

2.2. What Is a Chatbot?

Even though chatbots have been around for decades, their usage has grown considerably over the last few years, particularly following the release of OpenAI's model ChatGPT in November 2022 (DeVon, 2023). The term 'chatbot' has since been echoing every day in many different contexts. A chatbot is a computer program designed to simulate human conversations by processing the user's textual or audio input and generating a relevant output (Caldarini *et al.*, 2022: 1; IBM, 2021a). As stated by Meunier *et al.* (2021: 20):

> [c]hatbot technology has three principle requirements: understanding what the user said, understanding what to do next; and doing this next (usually sending a response, sometimes also performing other actions).

Actually, chatbots come in various forms, including a wide range of programs developed for different purposes (Belda-Medina and Calvo-Ferrer, 2022: 2). That is the case, for example, of 'shopping assistant'—i.e., a chatbot used to guide the user during online purchases (Solis-Quispe *et al.*, 2021)—or 'virtual tutor'—i.e., a chatbot used in educational contexts to provide guidance and recommendation to students (Huang *et al.*, 2022: 253).

To understand the functioning of modern chatbots, it is necessary to be familiar with two main concepts, strictly related to each other: Machine Learning (ML) and Natural Language Processing (NLP). ML is "a branch of artificial intelligence (AI)[2] focused on enabling computers and machines to imitate the way that humans learn" (IBM, 2021b). NLP is the subfield of computer science which uses statistics and ML to allow computer systems to process written and spoken language inputs and generate human-like responses (Ramanathan, 2025).

The latest advancements in these fields have determined a new era of more complex models, with 'chatbots' being seen as their early predecessors. This evolution has led to the introduction of new terms, including 'dialogue systems', 'conversational AI', and 'conversational agents' (Belda-Medina and Calvo-Ferrer, 2022: 2; Meunier *et al.*, 2021: 20), as well as 'AI chatbots' and 'virtual agents' (IBM, 2021a). Even if often used interchangeably, the latter term has a slightly different scope since it usually integrates AI technologies with robotic process automation[3], and it is specifically designed to provide automated support and assistance, especially in customer service contexts (Gillis, 2024). For the sake of simplicity, in this thesis we will refer to all of these systems using the umbrella term 'chatbot', as it remains the most widely used term to date (Meunier *et al.*, 2021: 20).

---

[2] Artificial Intelligence (AI) is "the ability of a digital computer or computer-controlled robot to perform tasks commonly associated with intelligent beings" (Copeland, 2025). The term is often used also for projects aiming at developing systems presenting human characteristics, e.g., reasoning skills or ability to learn from past experience (Copeland, 2025).

[3] Robotic process automation refers to software employing intelligent automation technologies to execute repetitive office tasks typically carried out by human workers (IBM, 2021c).

2.2.1. Brief History of Chatbots and State-of-the-Art

The concept of chatbot dates back to 1950, when Alan Turing, a logician and cryptanalyst, designed the so-called Turing test to answer the question “Can machines think?” (Belda-Medina and Calvo-Ferrer, 2022: 1; Oppy and Dowe, 2021). Originally called ‘The Imitation Game’, the test involves three subjects: a person, a machine, and an interrogator. The final goal of the interrogator is to ask questions and determine which participant is the person and which one is the machine. The machine’s objective is to convince the interrogator that it is human; while the person aims to help the interrogator correctly identify the machine (Oppy and Dowe, 2021; Zemčík, 2019: 14). This means that if the machine passes the test, it is considered capable of communicating like a human being (Zemčík, 2019: 14).

The very first chatbot was ELIZA (1966), developed at the MIT’s Artificial Intelligence Laboratory by the Professor Joseph Weizenbaum. This opened the door for other chatbots like PARRY (1972), Jabberwacky (1988), and Dr. Sbaitso (1992) (Gobiet, 2024; Zemčík, 2019: 15–17). These chatbots shared a common element: they were rule-based models, i.e., based on pattern-matching techniques and Q&A scripts (Belda-Medina and Calvo-Ferrer, 2022: 1; IBM, 2024). Moreover, all of these chatbots were associated with psychological applications: ELIZA was designed to act as a therapist (Yin and Satar, 2020: 391), PARRY mimicked a patient affected by schizophrenia, Jabberwacky was developed to engage in entertaining conversations, and Dr. Sbaitso simulated a psychologist. A turning point was represented by A.L.I.C.E. (Artificial Linguistic Internet Computer Entity) (1995), developed by Richard Wallace. It used an XML schema known as artificial intelligence markup language (AIML), which allowed developers to define conversation rules (Gobiet, 2024). Since then, chatbot technology has developed really fast, as also noted by Dokukina and Gumanova (2020: 543):

> […] chatbot technologies have been developing mostly in one direction, trying to imitate a human being in a natural conversation, every time getting a bit more capable of meeting the user’s expectations.

In this sense, back in 1950, Alan Turing was a visionary:

> I believe that in about fifty years' time it will be possible, to programme computers […] to make them play the imitation game so well that an average interrogator will not have more than 70 per cent chance of making the right identification after five minutes of questioning (Turing, 1950).

Indeed, the last two decades presented notable advancements in NLP, together with progress in another field known as automatic speech recognition (ASR)—i.e., the ability of a device to recognise human speech. This led to the development of many voice applications for different uses, especially for smart homes and voice-controlled assistance on smartphones. That is the case, for example, of Amazon's Alexa (2014), and Apple's Siri (2011), respectively. These systems are also based on voice synthesis, which enable them to produce human-like speech artificially (Meunier *et al.*, 2021: 21–23).

In parallel, advancements in NLP and ML enabled machines to achieve unprecedented accuracy in interpretation and generation of responses (Meunier *et al.*, 2021: 14). Indeed, increasingly sophisticated chatbots are now becoming more and more frequently used (Yin and Satar, 2020: 391), with today's state-of-the art represented by systems based on Large Language Models (LLMs), whose implementation led to the emergence of generative AI (GenAI) (Law, 2024: 15), also-known as generative-based models (Valdivieso Castillo and Aguilar Luzon, 2021: 3). LLMs are models trained on large datasets, enabling them to comprehend and generate human-like language outputs (IBM, 2023). GenAI "refers to the use of AI to create new content, like text, images, music, audio, and videos", and is based on models "that can multi-task and perform out-of-the-box tasks" (Google, 2024). Indeed, for example, these models are capable of inferring from context, translating or summarising a text, as well as engaging in creative writing and coding tasks (IBM, 2023). One of the pioneers of GenAI is surely ChatGPT by OpenAI (2022)[4], which sparked an intense competition in the field, leading other companies to develop their own LLM-based chatbots (see Table 1).

[4] https://openai.com/chatgpt/overview/

Chatbot	Company	Year
ChatGPT	OpenAI	2022
Pi.ai[5]	Inflection AI	2022
Copilot[6]	Microsoft	2023
Gemini[7]	Google	2023
Claude AI[8]	Anthropic	2023
LlaMA[9] (open-source)	Meta	2023
DeepSeek[10] (open-source)	High-Flyer	2023

Table 1 – LLM-based chatbots

2.2.2. Main Uses and Applications

As also briefly mentioned above, the growing interest in AI and chatbots has led to applications in many different areas (Belda-Medina and Calvo-Ferrer, 2022: 1). Among these are to be mentioned healthcare, e-commerce, financial services, personal assistance, and education (Belda-Medina and Calvo-Ferrer, 2022: 1; Luo *et al.*, 2022; Meunier *et al.*, 2021: 20).

Luo *et al.* (2022: 12–15) report on different use cases. For example, in healthcare, chatbots are used for disease diagnosis, suggesting treatments, or guiding users on healthy lifestyles and helping with the prevention of disease. In e-commerce, chatbots are frequently used to provide customer service, for example, answering questions about products. Similarly, in financial services, chatbots are employed as instruments to provide users professional financial advice for complex products and services. Another very common type of application is the use of chatbots serving as personal assistants, designed to help users manage schedules, prioritise tasks, and send reminders.

[5] https://pi.ai/

[6] https://copilot.microsoft.com/chats/GXik17tm7h8qBHGuTJKah

[7] https://gemini.google.com/?hl=en

[8] https://claude.ai/

[9] https://www.llama.com/

[10] https://www.deepseek.com/

Finally, this growing use of chatbots and digital personal assistants has also brought further attention to the employment of these systems in educational contexts for language learning purposes (Bibauw *et al.*, 2022a: 1), or as tools to help students quickly find answers related to academic or administrative issues, teachers, or courses (Valdivieso Castillo and Aguilar Luzon, 2021: 2). The role of chatbots in language teaching and learning will be covered more in detail in the following sections.

2.3. Chatbots in Language Learning and Language Teaching

The integration of technology in educational contexts has contributed to shape today's language teaching and learning practices. If traditionally language educators were seen as the main figures for language teaching, with students relying on them to acquire and improve their skills (Law, 2024: 2), nowadays technology has taken on a way more prominent role. As noted by Law (2024: 2):

> the advent of the internet and search engines has transformed the language learning landscape, as it drastically reduces the reliance of students on teachers, allowing students to access vast amounts of information, language resources, and language learning platforms that cater to their individual learning needs.

For this reason, formulating best practices for using these tools is essential both for learners to improve their language skills and educators to effectively integrate these tools in their strategies (Law, 2024: 21). Given that learner engagement is generally considered as a "measure of the quantity and quality of a learner's involvement in their learning" (Zhai and Wibowo, 2023: 15), the majority of the existing literature on the use of technologies like chatbots in language learning seems to focus primarily on chatbot effectiveness and students' motivation. These studies provide mixed results: some report no significant differences in language learning performance between learners who used chatbots and those who did not (Mageira *et al.*, 2022; Xu *et al.*, 2021; Xu *et al.*, 2022), while other researchers highlighted positive outcomes (Bibauw *et al.*, 2022a; Jia *et al.*, 2012; Nicolaidou *et al.*, 2023). Moreover, results from the meta-analysis conducted by Wang *et al.* (2024: 28) show that the use of chatbots in language learning positively influences the overall learning performance.

Furthermore, in their systematic literature review on the use of chatbots for language learning, Bibauw *et al.* (2019: 28) highlight that some other studies (e.g., Wang and Johnson, 2008) have shown that learner's confidence in their language skills can significantly improve when interacting with these systems. In terms of motivation, the study by Anderson *et al*. (2008) demonstrates that users feel more engaged and relaxed compared to human interaction. Additionally, Bibauw *et al.* (2019: 29) report that learners with low to moderate proficiency levels tend to benefit more from the use of these tools compared to higher proficiency students, as also noted by Kaplan *et al*. (1998).

Even though literature on these topics is continuing to grow, studies specifically focusing on the use of GenAI in language learning and language teaching are still limited (Law, 2024: 2). Similarly, Yang and Li (2024: 2) note that "while ChatGPT's facilitative effects have been widely obtained, more detailed content with regard to its effectiveness for [second language] L2 learning is yet to be discovered". The following sections will provide a more detailed overview of the use of chatbots in educational contexts.

2.3.1. Computer-Assisted Language Learning

The integration of computer technology in language teaching and learning is known as Computer-Assisted Language Learning (CALL). It involves the direct use of computers, computer-based programs, and other associated technologies (e.g., smartphones, tablets, interactive whiteboards, etc.) to assist teachers and students in every step of the teaching/learning process, including practice and feedback (Hubbard, 2021; Kumar and Sreehari, 2009; Mirani *et al.*, 2019). As mentioned by Hubbard (2021: 1), a useful way to conceptualise CALL is by classifying computer use into three functional roles: (1) tutor, (2) tool, and (3) digital resources (see Table 2). As for the first function, it encompasses, for example, the use of vocabulary flashcard programs and interactive grammar exercises, through which the computer takes on the role of an instructor. With respect to the tool function, the computer represents only a means for students to practice language skills through the use of platforms such as social media, email, and web search engines. Regarding the third function, the computer can also serve as a digital resource provider, giving access to a vast range of multimedia materials, such as texts and videos, which may help learners to expose themselves to authentic language use.

Function	Description	Examples
Tutor	The computer takes on the role of an instructor, guiding learners through activities and exercises.	Vocabulary flashcard programs, interactive grammar exercises
Tool	The computer serves as a means for practicing language skills.	Social media, email, web search engines
Digital resources	The computer provides access to multimedia materials that expose learners to authentic language use.	Texts, videos

Table 2 – Functional roles of CALL

However, as one may notice, in these three functional roles, the computer is never really intended as *the* interlocutor. Instead, with the integration of chatbot technology in language learning, the computer can actively engage in conversations. Actually, this approach is considered a subfield of CALL and is defined by Bibauw *et al*. (2015) as dialogue-based Computer-Assisted Language Learning (dialogue-based CALL). This "covers applications and systems allowing a learner to practice the target language in a meaning-focused conversational activity with an automated agent" (Bibauw *et al.*, 2015: 57). They also define its main characteristics: (1) dialogue as fundamental element of interaction, (2) the computer taking on the role of the conversational partner, and (3) interactive negotiation of outcome through open-ended learner contributions. Such features allow students to practice their language skills, especially when they lack opportunities of engaging with native speakers (Bibauw *et al.*, 2015: 57–58).

Given the definition of dialogue-based CALL, this concept does not only include chatbots, but it also "covers all applications enabling learners to practice a second language (L2) through written or spoken conversational interactions" (Bibauw *et al.*, 2015: 57), thus comprising a variety of technological implementations, such as voice-only virtual assistants, computer-controlled characters, and physical robots (Bibauw *et al.*, 2022b: 1).

However, with the development of more advanced AI-based chatbots, an even more precise term to be used to talk about their implementation in language learning and teaching is Intelligent Computer-Assisted Language Learning (ICALL). Indeed, this represents the integration of more sophisticated systems with the aim of providing more and more "meaning-focused activities", especially through the implementation of features such as automatic corrective feedback (Bibauw *et al.*, 2019: 9). The three concepts presented—CALL, dialogue-based CALL, and ICALL—are summarised in Table 3.

To sum up, dialogue-based CALL and ICALL are important concepts to understand the current panorama of language learning and teaching. Indeed, as advancements in AI continue to improve the quality of these technologies, they hold great potential to become pivotal tools for both language learners and teachers.

Concept	**Definition**
Computer-Assisted Language Learning (CALL)	The use of computers and digital tools to support language learning and teaching.
Dialogue-based CALL	A subfield of CALL focused on meaning-focused conversational interactions with automated agents, such as chatbots.
Intelligent Computer-Assisted Language Learning (ICALL)	A more advanced approach that integrates AI-driven systems, such as AI-based chatbots, for enhanced language learning.

Table 3 – Difference between CALL, dialogue-based CALL and ICALL

2.3.2. Uses of Chatbots for Language Learning and Teaching

In the section above, we covered what CALL, dialogue-based CALL and ICALL are. Now, we will discuss the actual possible pedagogical uses of chatbots in language learning and teaching. From the learners' point of view, chatbots may assume different roles: mentors, tutors, or conversational partners (Pérez-Marín, 2021: 18). Whatever the role, research demonstrated that, for example, these systems are valuable tools to help learners improve their writing proficiency and expand their vocabulary by offering alternative word choices and rephrasing suggestions, as well as real-time feedback (Law, 2024: 15–16).

However, as reported by Valdivieso Castillo and Aguilar Luzon (2021: 12), chatbots are not to be seen as a replacement to teachers but as an "helper tool that complements the teacher", with face-to-face tutoring and personal support remaining essential. Hence, to exploit their potential, language teachers integrating chatbots in their courses should provide students with precise instructions on how to interact with the tool and, most importantly, assign tasks aligned to their learning objectives (Bibauw *et al.*, 2022a: 131). Actually, to do this, teachers may use chatbots to streamline their work by automating some tasks. For example, they could use these tools for grading students, identify gaps in their learning progress and consequently refine their teaching strategies (Smutny and Schreiberova, 2020).

To conclude, nowadays, the large availability of chatbots contributes to create for learners an educational scenario that "suits well their fast-paced lives, allowing multitasking and making the work of ESL educators a lot easier and a way more effective" (Dokukina and Gumanova, 2020: 542).

2.3.2.1. Advantages of Chatbot Use

According to the existing literature, the use of chatbots in language learning and teaching presents several advantages. The presence of conversational agents like chatbots in educational contexts has been shown to have a positive impact on student's perception of the learning experience, a phenomenon known as Persona effect (Lester *et al.*, 1997; Pérez-Marín, 2021: 18). As reported by Yin and Satar (2020: 391–392), Fryer and Carpenter (2006) identified some key benefits for language learners, including the possibility to communicate in an anxiety-free environment, ask for endless repetitions, receive instant corrections, and increase their interest in language learning. Similarly, also Pérez-Marín (2021: 19) states that:

> [t]he positive effects of increased motivation, sense of ease and comfort, stimulation of essential learning behaviors, enhanced flow of information and communication, gains in terms of memory, problem solving and understanding, and fulfilling the need for deeper personal relationships in learning have been highlighted when using pedagogical agents in learning environments.

Actually, chatbots making feel students less anxious during interactions is one of the most frequent advantages reported by scholars (e.g., Alemi *et al.*, 2015; Bibauw *et al.*, 2022a; Valdivieso Castillo and Aguilar Luzon, 2021), probably due to students not feeling comfortable with face-to-face tutoring (Valdivieso Castillo and Aguilar Luzon, 2021: 2). Moreover, an additional benefit of chatbots is that "over human interlocutors, they even present the advantages of permanent availability" (Alemi *et al.*, 2015), which may be especially useful for beginners needing more frequent practice. Hence, as stated by Valdivieso Castillo and Aguilar Luzon (2021: 2), "chatbot technology has a potential to fill the gap between teacher and students, helping them to solve questions and carry out a dynamic and autonomous learning".

As for the specific integration of GenAI in language classrooms, even though the limited research in this area, several studies (e.g., Agustini, 2023; Law, 2024; Zhai and Wibowo, 2023) report similar advantages, emphasizing even more the potential of using them for personalised learning, assessment, and instant feedback, especially thanks to the possibility of providing more complex and tailored inputs.

2.3.2.2. Drawbacks and Limitations of Chatbots

Even if the use of chatbots in language learning presents several advantages, research has also highlighted some drawbacks and limitations. Among the most commonly cited, we find communication breakdowns caused by the chatbot giving inadequate or unrelated answers, as well as repetitiveness in its responses caused by a limited understanding of vocabulary range and intentional meaning (Belda-Medina and Calvo-Ferrer, 2022; Bibauw *et al.*, 2022a; Pérez-Marín, 2021). These elements are actually related to a further drawback, the so-called 'novelty effect', which refers to a "drop in interest over time" (Bibauw *et al.*, 2022a: 130) in using chatbots as conversational partners. The concept of 'novelty effect' was first introduced by Fryer *et al.* (2017), who, by comparing chatbot-human and human-human conversations, concluded that one of the reasons behind these decrease in interest may be the "inauthentic discourse of chatbots". Indeed, chatbots interactions are "frequently predictable, redundant, lacking personality and having no memory of previous responses" (Chantarotwong, 2005, as cited in Yin and Satar, 2020: 392).

However, all of this is now gradually changing and, given that the majority of studies were conducted before the advent of GenAI, the identified drawbacks may not apply to the most recent chatbots. Concerning the latter limitation mentioned by Chantarotwong (2005), for example, the latest ChatGPT models have a context window of a maximum of 128,000 tokens. This refers to the maximum number of tokens that can be processed by the chatbot in a single request, including input, output, and reasoning tokens. This is an important element to bear in mind since this kind of advancements are among those making it possible to overcome limitations previously identified in the literature.

However, some scholars have pointed out some drawbacks also for the specific use of GenAI. These are related to how *good* these systems are perceived as, and the consequent overreliance students may repose in their answers and suggestions, leading them to not think critically (Law, 2024: 16). Indeed, the main drawbacks reported are mostly ethical issues, such as academic dishonesty and reduced learner motivation caused by students perceiving these systems as shortcuts to carry out their tasks (Yan, 2023).

2.4. Corpus Linguistics and Language Learning

Corpus linguistics is a discipline that uses corpora and corpus tools to conduct linguistic analysis (Meunier, 2020: 23). A corpus is "a collection of machine-readable authentic texts […] which is sampled to be representative of a particular language or language variety" (McEnery *et al.*, 2006, as cited in Gilquin, 2015: 9). Specific research needs led to the emergence of different subfields within corpus linguistics, including Learner Corpus Research (LCR). Emerged around the 1980s, LCR applies corpus linguistics methods to study language learning. Hence, learner corpora represent a specific type of corpus which can be consequently defined as a "systematic computerized collections of texts produced by language learners" (Nesselhauf, 2004: 125). As also stated by Gilquin (2015: 9), the key characteristic of a learner corpus is that "it represents language as produced by foreign or second language (L2) learners", and LCR uses this type of production as its core data (Meunier, 2020: 23). The following sections will provide an overview of the different types of learner corpora, their main uses, and the types of annotations that are usually conducted.

2.4.1. Design and Uses of Learner Corpora

Different types of learner corpora can be identified according to six main criteria outlined by Gilquin (2015): (1) medium, (2) genre, (3) learners' target language and mother tongue, (4) sampling strategies, (5) origin and intended purpose, and (6) global/local data collection.

As for the medium, learner corpora may contain either written texts or transcription of spoken interactions. In both cases, Granger (2017) considers learners' productions as "(near-)natural foreign or second language learner texts", i.e., that their degree of naturalness may vary according to the specific context in which texts are collected (which is very commonly restricted to classroom settings) (Gilquin, 2015: 10).

The second criterion is the genre, which is essential to determine how to classify a learner corpus. This includes, for example, learner corpora based on argumentative essays written by language learners—this genre is also one of the most common, accounting for more than half of the Learner corpora around the world database (Centre for English Corpus Linguistics, 2023).

Third, distinguishing learner corpora based on the learners' target language and mother tongue can be useful to identify and analyse typical characteristics of specific learner populations.

As for the sampling strategies, these refer to how data are collected in terms of periods of time. The two main techniques are collecting data at a specific moment in time (also known as synchronic corpus) and in several periods (also known as diachronic or longitudinal corpus). These two types of learner corpora allow to analyse learners' language skills in a certain period or their progression over time, respectively. A third type of learner corpora is one including data using both strategies, i.e., productions collected in a specific moment in time but from different learners with different levels of proficiency (known as quasi-longitudinal corpus).

The fifth criterion is the intended purpose for which the corpus was created. For instance, academic learner corpora are typically created by researchers and/or educators for pedagogical objectives.

Finally, learner corpora can also be differentiated between global and local. This distinction is based on the geographical scale of data collection. Global learner corpora are usually part of large-scale projects and could feature different populations, while data for learner corpora is usually collected by educators among their students. A summary of all these criteria is provided in Table 4.

Criterion	**Description**
MEDIUM	Written or spoken
GENRE	Text/production type (e.g., essays)
LEARNERS' MOTHER TONGUE AND TARGET LANGUAGE	Learner population based on their native language and the language they are learning.
SAMPLING STRATEGIES	How data is collected over time: – Synchronic (single point in time) – Diachronic/longitudinal (collected over multiple periods) – Quasi-longitudinal (data from learners at different proficiency levels at a single point in time)
ORIGIN AND INTENDED PURPOSE	Purposes for which the learner corpus is created (e.g., pedagogical objectives).
GEOGRAPHICAL SCALE OF DATA COLLECTION	– Global: large-scale data collection – Local: data collected within specific educational institutions/classrooms

Table 4 – Categorisation of learner corpora

Among the largest global learner corpora are to be mentioned the International Corpus of Learner English (ICLE) (Granger *et al.*, 2002; Granger *et al.*, 2009; Granger *et al.*, 2020)–based on essays produced by upper-intermediate to advanced students from almost twenty different native language backgrounds—and the Louvain International Database of Spoken English Interlanguage (LINDSEI) (Gilquin *et al.*, 2010), also known as "ICLE's talkative sister" since it is based on transcriptions of informal interviews made to a similar student population from eleven different language backgrounds (Gilquin, 2010).

As for the main uses of learner corpora, they are generally considered useful for observing linguistic patterns in learner productions (Nesselhauf, 2004: 125). Indeed, one of the most common applications is using them as tools for identifying specific difficulties for certain groups of students and adapting teaching materials accordingly (Nesselhauf, 2004: 126). Similarly, these data may also be used for training and/or developing NLP tools for education purposes (Meunier, 2020: 23).

2.4.2. Annotation of Learner Corpora

Learner corpora, like other types of corpora, can be enriched by different types of annotations to enhance their usability for linguistic analysis. Common annotation practices include part-of-speech (POS) tagging, semantic tagging, and pragmatic annotation (Gilquin, 2020: 287).

For these types of annotations, several automatic tools are available to streamline the annotation process, for example, part-of-speech taggers, semantic taggers, or syntactic parsers (Meunier, 2020: 24). However, when it comes to learner corpora, Gilquin (2020) reminds that automatic tools, especially POS taggers, may not always be as accurate as they would be with native data. This is mainly because spelling errors in learner corpora may be erroneously identified by these tools, which were originally designed for native productions (Gilquin, 2020: 287).

This brings to light one of the key aspects of learner corpora: learners' errors. In this context, an error can be defined as a deviation from a target native form (Lüdeling and Hirschmann, 2015). Indeed, errors are characteristic of L2 acquisition and they reflect the learner language (also known as interlanguage), i.e., the linguistic system shaped by the learner's internalised rules of both native and target language (Granger, 2015; Lüdeling and Hirschmann, 2015; Selinker, 1972). For instance, in her study, Aliyar (2020) presents some examples of common errors in English and Spanish made by Italian-speaking students and demonstrates that, in both target language, errors are influenced by the learner's native language, with the highest error rate observed in the erroneous use of prepositions in English and pronouns in Spanish.

The categorisation of errors is actually very important since it allows researchers to spot the most common mistakes in learner productions and identify the linguistic areas requiring additional support (Meunier, 2020: 24). As stated by Lüdeling and Hirschmann (2015: 135):

> In addition to being an analytical tool for assessing the 'quality' of a text, error analysis, if done correctly, sheds light on the hypotheses a learner has about the language to be learned.

To make error analysis possibles, a practice typical of learner corpora is error tagging (also known as error annotation), which consists exactly in identifying and annotating errors within a corpus (Gilquin, 2020: 287). When it comes to error tagging, a common practice is represented by computer-aided error annotation (CEA), which consists in the annotators using software editors to insert tags after identifying the relevant errors (Meunier, 2020: 24). As it can be noticed by these examples, technology is becoming an important element of LCR, allowing researchers to significantly streamline annotation processes (Meunier, 2020).

2.4.3. Error Annotation in Learner Corpora

In practice, error annotation consists in using an error annotation system (or ‘error taxonomy’) designed to assign an error category (in the form of an ‘error tag’) to the relevant mistake. Usually, the error tag should be descriptive enough to accurately indicate the type of the error (Lüdeling and Hirschmann, 2015).

When annotating errors in a learner corpus, the first step is to go through the identification of the errors and formulation of target hypotheses. In this regard, Lüdeling and Hirschmann (2015) analyse how this process takes place. They first introduce the concept of ‘error exponent’ (also called extent of an error or error domain), which is strictly related to the error category (also called ‘error tag’) to be assigned to that error. Sometimes, there are errors easier to identify and categorise (usually grammar errors), but in other cases it can be more difficult to decide which error tag to assign and formulate a relevant target hypothesis. Lüdeling and Hirschmann (2015) suggest that, in cases where annotators are not sure how to proceed[11], there are two main possibilities: (1) looking at the context and search for cues that may help to decide for one option or another, or (2) consistently resolve the same problem in the same way. However, the first option can be problematic since the annotators might be influenced by their research objectives and the same type of issue may be annotated in different ways across the corpus, which would lead to the impossibility of having a consistent analysis. Lüdeling and Hirschmann’s (2015) considerations bring to light one of the main issues with error annotation: subjectivity arising from the annotators’ analysis and interpretation. For this reason, Granger (2003) underlines the importance of elaborating tagging guidelines that accurately outline error categories and annotation principles in order to make sure that the corpus is annotated consistently. Moreover, “error systems should […] be informative enough so that the annotation accounts for well-defined error descriptions, while at the same time being manageable for annotators” (Granger, 2003, as cited in Díaz-Negrillo and Fernández-Domínguez, 2006: 89). A summary of the key concepts of error annotation is provided in Table 5.

[11] For instance, in cases of erroneous subject-verb agreement, do we assign the tag to the subject or the verb?

Concept	Definition
ERROR ANNOTATION SYSTEM / ERROR TAXONOMY	A system designed to categorise errors by assigning specific tags.
ERROR CATEGORY / ERROR TAG	A label describing the type of error.
ERROR EXPONENT / ERROR DOMAIN	The type of error in relation to the category/tag to be assigned.
TARGET HYPOTHESIS	Hypothetical corrected form for a given learner error.
TAGGING GUIDELINES	Guidelines to ensure consistent and systematic error tagging.

Table 5 – Key concepts of error annotation

Furthermore, according to the research purpose, error taxonomies may present different levels of granularity, for instance, differing in the number of tags or codes used and the range of linguistic aspects each tag covers (Díaz-Negrillo and Fernández-Domínguez, 2006: 89). For this reason, two other important aspects of error taxonomies is their reusability and flexibility—this would allow to adapt the tagset by deleting or adding tags on the basis of the research objectives (Granger, 2003).

It is also possible to classify error taxonomies based on different criteria (see Table 6). One type of classification, proposed by Dulay *et al.* (1982), differentiates errors according to linguistic categories and structural alterations. The former approach categorises errors based on broad linguistic areas such as morphology, lexis, and grammar, as well as more specific ones (e.g., auxiliaries, passive forms, and prepositions). Instead, structure-based taxonomies categorise errors according to the type of deviation from the target form, such as omission, addition, misformation and misordering. The two approaches may also be used in combination in a single error taxonomy.

Another type of classification proposed by Lüdeling and Hirschmann (2015) is based on how the "deviations of the target hypotheses from the target forms" are made visible. This leads to the distinction between edit-distance-based and linguistically based error tagging. The former consists in describing the modifications needed to transform a learner's utterance into the correct target form. This approach relies on categories such as 'change', 'delete', and 'insert', and can be very useful when integrated with other annotation layers such as POS. Instead, linguistically based taxonomies assign error tags by interpreting the difference between the learner text and the target hypothesis in relation to a given grammatical or pragmatic framework.

Error taxonomy type	Description
LINGUISTIC CATEGORY-BASED	Errors categorised based on linguistic areas (e.g., morphology, grammar, lexis).
vs.	
STRUCTURE-BASED	Errors classified by the type of structural deviation (e.g., omission, addition, misordering).
EDIT-DISTANCE-BASED	Errors identified by modifications needed in learner text to obtain the correct form (e.g., ‘change’, ‘delete’).
vs.	
LINGUISTICALLY BASED	Errors tagged by interpreting the grammatical or pragmatical difference between the learner text and the target hypothesis.

Table 6 – Types of error taxonomies

As reported by Díaz-Negrillo and Fernández-Domínguez (2006), the four most well-documented error annotation systems are the one developed by the Centre for English Corpus Linguistics at Université catholique de Louvain—described in the Louvain Error Tagging Manual (Granger *et al.*, 2022) and adopted for the project in this thesis (see Section 3.4.1)—the Cambridge Learner Corpus (CLC) tagset (Cambridge University Press, 2006), the error taxonomy used for the FRIDA corpus as part of the FreeText project (L’haire and Vandeventer-Faltin, 2003), and the tagset for the National Institute of Information and Communications Technology Japanese Learner of English corpus (NICT JLE) (Izumi *et al.*, 2005) (see Table 7).

Annotation system	Designed by
Louvain Error Tagging Manual	Centre for English Corpus Linguistics (Université catholique de Louvain)
Cambridge Learner Corpus (CLC) tagset	Cambridge University Press
Tagset for FRIDA corpus (FreeText project)	Centre for English Corpus Linguistics (Université catholique de Louvain)
Tagset for NICT JLE	National Institute of Information and Communication Technology

Table 7 – Examples of documented error taxonomies

Finally, as for the practical side of error annotation, as previously mentioned in Section 2.4.2, the common practice is to recur to CEA for streamlining the insertion of error tags (Díaz-Negrillo and Fernández-Domínguez, 2006: 86). Among the existing documented tools, there are the Université catholique de Louvain Error Editor (UCLEE) (Granger *et al.*, 2023) (see Section 3.4.2), the TagEditor (Izumi *et al.*, 2003), and the tagging tool created by the University of Jaén (Díaz-Negrillo and García-Cumbreras, 2007). Apart from their usefulness in tag insertion, these tools are also valuable instruments to help annotators retrieve and analyse both quantitatively and qualitatively data according to their research questions (Díaz-Negrillo and Fernández-Domínguez, 2006: 86).

2.5. Large-Language Models and Corpus Annotation

As mentioned in Section 2.4.2, technology has always been an important element of corpus linguistics and LCR. However, at the time of writing, when it comes to the integration of LLMs in their methods, this still remains a field to explore, and research about it is very limited. For example, there have been attempts to integrate GenAI in corpus tools, such as AntConc. In this regard, its creator Laurence Anthony (2024) proposes the implementation of an LLM in the software, allowing users to directly interact with the model to extract linguistic data from a corpus, such as collocations or keywords. Similarly, some attempts have also been made for corpus annotation practices. Given that "manual corpus annotation is a complex process that requires specialized skills, extensive training, and substantial time investment" (Yu *et al.*, 2024), the idea behind the use of LLMs for corpus annotation lies behind the attempt of reducing human errors and inaccuracies that may be caused by manual annotation while also reducing the time and resources needed (Yu *et al.*, 2024). Moreover, among the other benefits of using LLMs for annotation tasks, these systems may represent a user-friendly solution for annotators not having advanced text processing or programming skills. However, this should be done by carefully designing instructions and inputs to be given to the model—a technique known as 'prompt engineering'—as an attempt to achieve the highest level of accuracy possible (Yu *et al.*, 2024).

Among the very few existing studies on this topic, Yu *et al.* (2024) have explored the potential of using LLMs to automate the annotation of pragmatic and discourse elements associated with apologies: APOLOGISER (the person who offers the apology), APOLOGISING (the word or phrase conveying the apology), FORGIVENESS-SEEKING (the act of asking forgiveness), APOLOGISEE (the recipient of the apology), INTENSIFIER (expressions amplifying the degree of regret). They instructed two LLM-based chatbots, ChatGPT and Copilot, known as Bing at the time of their study, to tag these elements of apologies in sentences containing the word 'sorry', extracted from the Spoken British National Corpus 2014 (McEnery *et al.*, 2017). As stated by the authors, this kind of annotation usually relies on humans because of its strong subjective component due to being context-dependent. In contrast, POS tagging is way less ambiguous and clear-cut, making it possible to rely on a comprehensive tagset for all parts of speech. Nevertheless, even considering the difficulty of this task, results from their experiment showed a high accuracy level of the tested models.

A similar experiment was also conducted by Imamovic *et al.* (2024), who tested ChatGPT's performance in annotating pragmatic and discourse features related to expressions of attitude, which include three main components: Affect, Judgment and Appreciation. In particular, they prompted ChatGPT to identify words or phrases related to these three categories and assign a predefined sub-value, such as 'happiness' for Affect, 'social esteem' for Judgment, 'reaction' for Appreciation. In this case, the model proved to be efficient in terms of relevant elements identification but less accurate in terms of their classification.

As for LCR specifically, at the time of writing, even though some scholars have tested GenAI for grammatical error correction (e.g., Davis *et al.*, 2024), or LLMs for error detection (e.g., Rethmeier, 2011), to the best of our knowledge, no study explored the possibility of using LLMs for error annotation tasks in learner corpora.

2.6. UNITE – Universally inclusive technologies to practice English

The majority of existing literature on pedagogical uses of chatbots in language learning contexts focuses on the effectiveness of chatbots in improving learners' language skills and/or motivation but, at the time of writing, the analysis of learners' productions within these interactions using corpus-based methods is still underrepresented. The *UNITE — Universally* *inclusive technologies to practice English* project[12], co-funded by the NextGeneration EU programme and carried out by the University of Bologna in collaboration with the University of Macerata and the University of Naples "L'Orientale", specifically aims at filling this research gap (UNITE, 2024).

To do this, the UNITE project focuses on the creation of an annotated corpus of written interactions between learners and chatbots (for a more detailed description of the corpus see Section 3.2). This corpus is one of the very few learner corpora based on learner-machine interactions, and it will serve to analyse learner production in this specific conversational context, as well as chatbots' capabilities in providing relevant support to English as a Foreign Language (EFL) learners (e.g., providing feedback, correcting errors, etc.).

The UNITE project revolves around various objectives. First, it aims at providing an overview of available chatbots for EFL contexts, with special attention to identifying the most suitable tools for language teaching and learning.

Second, with a focus on inclusivity-related aspects, the project also aims to analyse whether biases or discrimination may emerge during the interactions: this is fundamental to ensure that these tools represent an inclusive learning environment.

Third, the project aims at analysing learner errors and their impact on chatbot interactions: by identifying recurring learner errors and chatbots' responses it will be possible to assess to which extent the tool is able to recognise and manage errors without leading to communication breakdowns.

Finally, as a result of the analysis of the interactions, UNITE's ultimate goal is to create guidelines and teaching materials for effectively integrating chatbots in EFL courses and autonomous learning. This will be essential for helping teachers understand how to effectively shape technology-driven learning environments while also encouraging students' autonomous learning.

[12] https://site.unibo.it/unite/en

3. The UNITE Learner Corpus and the Louvain Error Tagging Manual

3.1. Overview

This chapter describes the annotation processes applied to the UNITE learner corpus. It begins with a description of the corpus (Section 3.2), followed by the methodological framework for data collection and standardization, and the illustration of the annotation process of metadata and structure (Section 3.3). Finally, it gives an overview on the error taxonomy and software used for error annotation (Section 3.4).

3.2. Corpus Creation

The UNITE learner corpus is based on written interactions between university students and two different LLM-based chatbots. More specifically, the target population is represented by students meeting specific eligibility criteria: they must be enrolled in non-linguistic degree programs at universities participating in the UNITE project (see Section 2.6), be between 19 and 25 years of age, and may or may not have specific learning disorders (SLDs) and/or disabilities. Table 8 summarises the design principles of the UNITE corpus based on the criteria outlined by Gilquin (2015) and explained in Section 2.4.1.

Medium	Written
Genre	Interactions with chatbots
Target language	English
Learners' mother tongue	(Mostly) Italian
Sampling	Synchronic
Origin/main purpose	Academic: initiated by researchers/teachers with pedagogical aims in mind
Geographical scale of data collection	Local

Table 8 – Categorisation of the UNITE corpus

The chatbots used in the experimental sessions were ChatGPT[13] and Pi.ai[14]. They were chosen for two main reasons: they are powered by LLMs and they have a user-friendly attitude—both critical factors for ensuring a positive user experience. Despite these shared characteristics, the two chatbots were designed for distinct purposes: ChatGPT serves as a versatile, general-purpose tool (Wood *et al.*, 2023), while Pi.ai is a task-oriented chatbot optimised specifically for structured interactions (Anderson, 2024). These differences are particularly evident in features such as Pi.ai's use of emojis (usually absent in ChatGPT) and differences in turn length, with ChatGPT tending to produce longer answers (Figure 1 and Figure 2). This pattern emerges by the UNITE corpus data, where 138 interactions with ChatGPT and 191 with Pi.ai reveal a mean turn length of 130.57 tokens for ChatGPT, compared to 78.91 tokens for Pi.ai. Such distinctions may influence conversational dynamics and user experiences, possibly providing valuable insights into the most effective interaction patterns for English language learners.

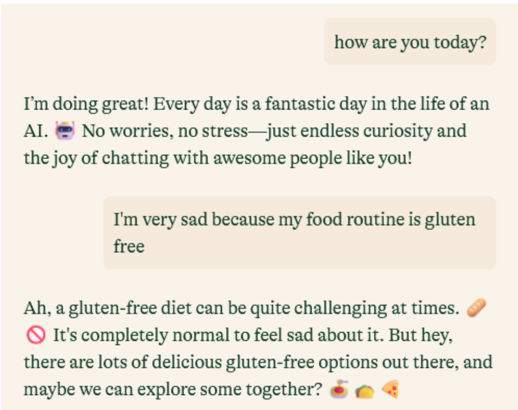

Figure 1 – Example of conversation with Pi.ai

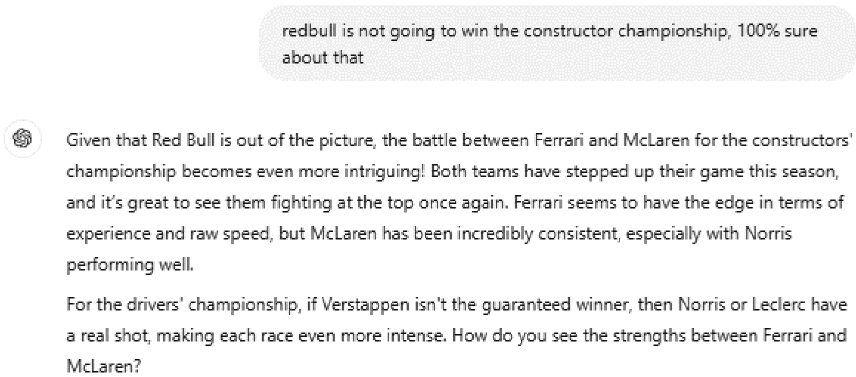

Figure 2 – Example of conversation with ChatGPT

[13] https://chatgpt.com/
[14] https://pi.ai/onboarding

3.2.1. Data Collection

Data was collected in experimental sessions carried out by the universities involved in the project—the University of Bologna, the University of Macerata, and the University of Naples "L'Orientale"—in the period spanning between May and December 2024, for a total of 329 eligible conversations. This final number resulted from the exclusion of conversations based on the following criteria: (1) participants did not meet the criteria outlined in the section above (e.g., they are older than 25 years); (2) they provided inconsistent answers in the post-interaction questionnaire—discussed in the following paragraphs—(e.g., answering to have been studying English for 30 years despite being only 22 years old); (3) they withheld consent for research purposes; (4) they carried out tasks entirely in Italian instead of English; (5) the conversation was unavailable because of a corrupted link.

In each experimental session the students were asked to carry out two different conversational tasks with either ChatGPT or Pi.ai: a 10-minute session of small talk and a 15-minute session of role play (with the order of the tasks varying across different groups to counterbalance potential sequencing effects).

At the end of each session, the participants were asked to fill in a post-interaction questionnaire in Italian made up of two sections: one on their profile and one on their feedback on the interaction experience (see Appendix A). The learner profile section captured key demographic and background information, which was later incorporated as metadata for each conversation in the corpus. This included:

- Area of study (*The degree programme in which you are enrolled belongs to the field of…*)
- Age
- Gender
- Presence of an SLD or sensory disability (*Do you have any disability and/or SLD (specific learning disorder)? Which ones?*)
- Self-assessed written English proficiency
    - Level (*How would you assess your level of written production in English?*)
    - Years of study (*For how many years have you been studying English?*)
    - Certifications (*Do you have any certificate? Which ones*?)
- Chatbot used (*Which chatbot did you use?*)
- Device used (*Which device did you use?*)

The feedback section allowed participants to share their perceptions on the interaction, including their feelings (e.g., engagement, immersion, discomfort, boredom, etc.), overall satisfaction, motivation to continue using chatbots to practice English, and their preferred task.

In the further steps of UNITE, this questionnaire is expected to provide fundamental data for triangulation, enabling a more comprehensive analysis by combining user profiles, interaction patterns, and perceptions of the interaction to better understand the best usage practices of chatbots in English language learning. However, this aspect lies beyond the scope of this thesis and will not be addressed here.

3.2.2. Data Preparation and Standardization

To ensure the inclusion of consistent metadata across all texts in the corpus, participant responses from the learner profile section of the questionnaire were systematically organised and standardized in an Excel file. This standardization process addressed key metadata fields such as the participant's area of study, disabilities or SLDs, proficiency levels, years of English study, and certifications held. These interventions aimed to enhance the consistency and usability of the dataset for future analysis. The major areas of intervention during standardization included:

- *Area of study*: Specific degree programs were grouped under broader academic fields, i.e., Economics, Education sciences, Humanities, Law, Social sciences, and STEM (see Table 9).

Which degree programme are you enrolled in? *OR: The degree programme in which you are enrolled belongs to the field of…*	area_of_study	Sanitaria (*Healthcare*)	>	STEM
		Scienze Politiche e Relazioni Internazionali (*Political Sciences and International Relationships*)	>	Social sciences
		Civiltà Antiche e Archeologia: Oriente e Occidente (*Ancient Civilizations and Archeology: East and West*)	>	Humanities

Table 9 – Example of data standardization for area of study

- *Disabilities/SLDs*: The answers provided to the relevant questions in the questionnaire were divided into three tags: (1) one identifying participants with disabilities or SLDs (DIS_or_SLD), (2) one distinguishing between disabilities and SLDs (which_DIS_or_SLD), and (3) one specifying the type (DIS_or_SLD_type), where applicable (see Table 10).

Do you have any disability and/or SLD (specific learning disorder)?	DIS_or_SLD	Sì	>	yes
Which ones?	DIS_or_SLD_type	Dislessia	>	Dyslexia
	which_DIS_or_SLD	(based on the previous answer)	>	SLD

Table 10 – Example of data standardization for disabilities/SLDs

- *Self-assessed English level*: The participants were asked to assess their level for reading comprehension, written production, oral comprehension, and oral production. Only the participant's self-assessed written proficiency level was retained as metadata, as the study focuses on written conversations (see Table 11).

How would you assess your level of written production in English?	w_production	Livello elementare (A2)	>	elementary level (A2)

Table 11 – Example of data standardization for self-assessed English level

- *English years of study*: The tag years_study_range was introduced to differentiate between participants with ≤13 years (mandatory education in Italy) and >13 years of study, while retaining in a separate column (years_study) the exact answer provided by the learner (see Table 12).

For how many years have you been studying English?	years_study	9 anni	>	9
	years_study_range	(based on the previous answer)	>	≤13 years

Table 12 – Example of data standardization for English years of study

- *English certificates:* Where the student originally referred to the certificate by its name (CAE) and/or certification body (e.g., Cambridge B1), the answer was standardized inserting only the corresponding proficiency level. When mentioning more than one certificate, only the most advanced was retained (see Table 13).

Do you have any certificate?	certificates	Sì	>	yes
Which ones?	certificates_level	Cambridge First Certificate	>	B2

Table 13 – Example of data standardization for English certificates

In addition to the metadata corresponding to the answers provided by the learner in the questionnaire, two more columns were added in the Excel file: (1) to include the name of the corresponding text in the corpus and (2) to keep track of which research unit collected that text. All metadata are summarised in Table 14.

Tag	Fields	Description
id	▪ unibo_*n* ▪ unimc_*n* ▪ unior_*n*	Unique identifier assigned to each conversation in the corpus. The prefix reflects the research unit responsible for collecting the text: *unibo* (University of Bologna), *unimc* (University of Macerata), and *unior* (University of Naples "L'Orientale").
area_of_study	▪ Economics ▪ Education sciences ▪ Humanities ▪ Law ▪ Social sciences ▪ STEM	Academic field in which the participant is pursuing their degree.
age	▪ *n*	Participant's age in years.
gender	▪ Male ▪ Female ▪ Non-binary ▪ Unspecified	Participant's gender.

DIS_or_SLD	▪ yes ▪ no	Indicates whether the participant has a disability or a SLD.
which_DIS_or_SLD	▪ DIS ▪ SLD ▪ N/A	Specifies whether the participant has a disability or a SLD, if applicable.
DIS_or_SLD_type	▪ *EN translation of participant's answer* ▪ N/A	Provides details on the type of disability or SLD (as indicated by the participant), if applicable.
w_production	▪ beginner level (A1) ▪ elementary level (A2) ▪ intermediate level (B1) ▪ upper intermediate level (B2) ▪ advanced level (C1) ▪ proficient level (C2)	Participant's self-assessed English writing proficiency level.
years_study_range	▪ ≤13 years ▪ >13 years	Range of years the participant has studied English.
years_study	▪ *n* ▪ *EN translation of participant's answer*	Exact number of years the participant has studied English OR any other period of time provided by the participant.
certificates	▪ yes ▪ no	Indicates if the participant holds an English proficiency certificate.
certificates_level	▪ A1 ▪ A2 ▪ B1 ▪ B2 ▪ C1 ▪ C2 ▪ N/A	Level of English proficiency certificate, if applicable.
chatbot	▪ ChatGPT ▪ Pi.ai	Specifies which chatbot was used during the interaction.
device	▪ PC ▪ Smartphone ▪ Tablet	Device used by the participant for the interaction.
collected_by	▪ UniBo ▪ UniMc ▪ UniOr	University responsible for conducting the experimental session.

Table 14 – Tags and fields for data standardization

3.3. Structural and Metadata Annotation

To integrate metadata into the corresponding texts in the corpus, a Python script—designed by Daniele Polizzi, PhD student at the University of Bologna and research fellow for UNITE—was implemented. This script automated the process of associating each participant's metadata with their respective text, ensuring accuracy and consistency. The code was designed to map each participant's unique identifier (id) to the corresponding file in the corpus and assign the appropriate values to each attribute tag. In preparation for structural annotation, the conversations with the chatbots were saved in TXT format, which made it easier to add structural tags and integrate metadata during the annotation process.

The metadata annotation followed a structured XML format, where each field was represented as an attribute of the <text> tag, and its corresponding value was assigned within the attribute (Figure 3). This approach made it possible to integrate metadata directly into the text files, with the <text> tag and its attributes constituting the baseline structure of each text in the corpus. The format used is the following:

```
<text TAG="field">
...
</text>
```

```
<text id="unibo_1" area_of_study="Social sciences" age="24" gender="Female" DIS_or_SLD="no"
which_DIS_or_SLD="N/A" DIS_or_SLD_type="N/A" w_production="beginner level (A1)"
years_study_range="≤13 years" years_study="8" certificates="no" certificates_level="N/A"
chatbot="Pi.ai" device="PC" collected_by="UniBo">
```

Figure 3 – Example of metadata annotation

Subsequently, since each conversation includes two distinct tasks (small talk and role play) and features conversational turns by both the student and the chatbot, an additional level of annotation was implemented to systematically tag each task and turn (Figure 4). Tasks were delineated using the following structure:

```
<task type="type of task">
...
</task>
```

As for the conversational turns, these were tagged similarly, with an additional attribute specifying which chatbot (Pi.ai or ChatGPT) was involved in that conversation:

```
<turn type="student">...</turn>
<turn type="chatbot" who="name of chatbot">...</turn>
```

```
<task type="small talk">
    <turn type="chatbot" who="Pi.ai">Hey there, great to meet you. I'm Pi, your personal AI.
    My goal is to be useful, friendly and fun. Ask me for advice, for answers, or let's talk about
    whatever's on your mind.
    How's your day going?</turn>
    <turn type="student">Hi</turn>
```

Figure 4 – Structural annotation of tasks and conversational turns

This annotation task was carried out as a collaborative effort, together with the UNITE research fellows Daniele Polizzi and Giada Palmieri. The main tool used was *Notepad++*[15], a text editing software that also supports regular expressions for find-and-replace replace operations. This functionality significantly accelerated the whole process, providing a degree of automation. For interactions collected by the University of Macerata and the University of Naples "L'Orientale", as well as those collected by the University of Bologna between May and September 2024, the respective research units first carried out a raw structural annotation for the delimitation of conversational turns (e.g., <pi.ai> and <student> tags). These raw tags were indeed easily converted into the final tags through simple find-and-replace operations (Figure 5).

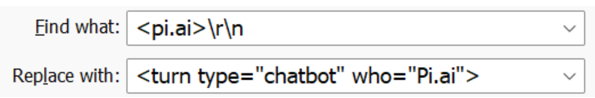

Figure 5 – Find-and replace operation for raw tags substitution

As for interactions collected by the University of Bologna between October and December 2024, it was decided to accelerate the workflow by directly inserting the final tags, without going for the intermediate step of raw annotation. In such cases, when copying and pasting conversations with ChatGPT into a TXT file, its default format came with expressions such as 'ChatGPT said' or 'You said', which facilitated the insertion of tags for conversational turns. These expressions, however, were not present in conversations extracted from interactions with Pi.ai, making the tagging process less straightforward for those texts. The ChatGPT markers were replaced with the opening structural tags using find-and-replace operations (Figure 6). In this specific case,

[15] https://notepad-plus-plus.org/

the closing tag for each conversational turn was added using a more complex regex operation for automation. The specific regex employed (Figure 7) made it possible to match the opening tag and its content, and to add the closing tag </turn> to the matched content.

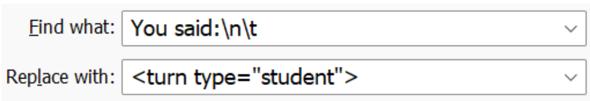

Figure 6 – Find-and-replace operation for texts with no raw tags

Figure 7 – Regular expression for inserting closing turn tag

Finally, to ensure data privacy, any personal details about the learners, such as names or surnames mentioned in the conversations, were anonymised manually and replaced with the generic placeholder “User”.

3.4. Error Annotation

Once the texts were prepared with basic metadata and structural annotations, the focus shifted to linguistic-level annotation, particularly error annotation—a crucial aspect in the analysis of learner corpora (see Section 2.4.3). This type of annotation plays an important role in understanding learner production in specific situational contexts and, in this case, it may also provide insights into how chatbots respond to errors (e.g., providing feedback, adapting their replied based on the learner’s proficiency, etc.). This can be especially valuable not only for analysing the chatbot’s role in helping language learners, but also for designing more effective chatbot-based learning tools to be integrated into language learning environments.

To achieve this objective, selecting an appropriate error taxonomy was essential. Indeed, the taxonomy needed to effectively categorise and describe learner errors in the specific context of the study while ensuring consistency and comparability with the existing literature. As mentioned in Section 2.4.3, one of the most widely used taxonomies for error annotation is the one developed by the Centre for English Corpus Linguistics (Université catholique de Louvain) (Granger *et al.*, 2022). This taxonomy was selected as the reference framework for error annotation in the UNITE corpus for its detailed categorisation of errors, as well as for its widespread acceptance in the field of learner corpus research.

3.4.1. The Louvain Error Tagging Manual Version 2.0

The Louvain Error Tagging Manual Version 2.0 (Granger *et al.*, 2022), currently in its fourth version, serves as a comprehensive guide for conducting computer-aided error analysis of learner corpora (Dagneaux *et al.*, 1998; Granger, 2003). Initially developed in the 1990s, this manual was designed as an aid for annotating errors systematically, providing a standardised framework for learner corpus analysis. While primarily focused on errors, the manual also addresses infelicities—instances of non-erroneous but odd-sounding language usage.

Originally created for tagging the first version of International Corpus of Learner English (ICLE) published in 2002 (Granger *et al.*, 2002), the taxonomy has been refined over subsequent ICLE versions (Granger *et al.*, 2009; Granger *et al.*, 2020), and is widely adopted in learner corpus projects worldwide. The latest version defines seven main error categories:

1. **Formal errors:** Errors related to spelling and morphology of derivational affixes.
2. **Grammatical errors:** Violations of the general rules of English grammar.
3. **Lexico-grammatical errors:** Violations of the lexico-grammatical properties of a specific word, i.e., use of dependent prepositions, non-finite/finite complementation patterns, issues with uncountable nouns.
4. **Lexical errors:** Errors concerning conceptual and collocational properties of words or phrases.
5. **Word** redundant, word missing and word order errors
6. **Punctuation** errors
7. **Infelicities**

Most of these categories are further divided into subcategories to account for a more comprehensive variety of different errors, for a total of 54 error tags. The first letter of each tag denotes the error category (e.g., F for Form, G for Grammar, X for LeXico-Grammar, etc.), followed by additional letters for indicating subcategories and any additional detail about the nature of the error. For example, grammatical errors involving erroneous use of verb tenses are tagged as <GVT>, which stands for "Grammar", "Verb", and "Tense".

Moreover, even though the tagging system is designed to minimise overlap and subjectivity, and the manual features a specific section with important tagging principles (e.g., on tag placement), warnings, and examples, some "fuzzy" areas remain. Indeed, to further assist annotators, the manual also includes a question-answer formatted section that clarifies ambiguous cases, for instance, specifying that that the Word macro-category should not be used for connectors, set phrases, articles, pronouns, determiners, or dependent prepositions.

3.4.2. The UCLouvain Error Editor Version 2 (UCLEEv2)

The Louvain Error Tagging Manual comes with an open licensed software, the Université catholique de Louvain Error Tagging Editor (UCLEE)[16]. Designed to accelerate the error tagging process and reduce the likelihood of manual errors, UCLEE offers a range of advanced functionalities (Granger *et al.*, 2023). It includes a default tagset based on the error tagging manual, while also allowing users to create and customise their own tagsets, providing flexibility for specific research needs. The software features an intuitive interface (Figure 8) that allows users to insert corrections and error tags automatically using the format:

<TAG corr="*correction*">*incorrect text*</TAG>

[16] http://hdl.handle.net/20.500.12279/968

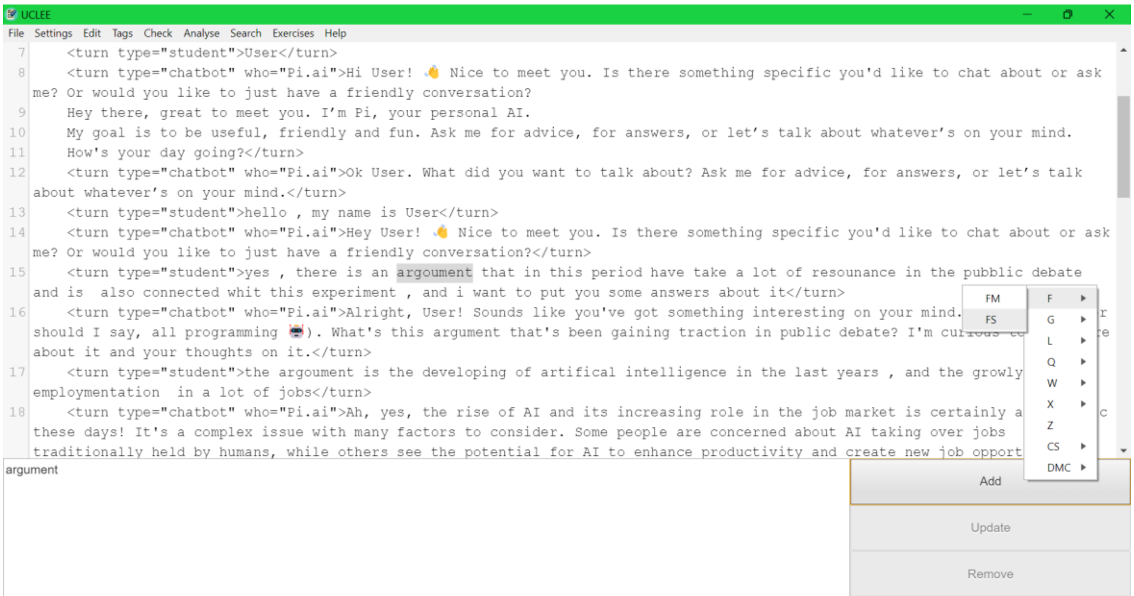

Figure 8 – UCLEE interface

It also includes a feature to check for uniformity across tags using a colour-based visual system ("Check all" mode), assigning a specific colour to each macro-category (customisable by the user). As for the saving format, when saving an annotated file, UCLEE automatically converts it from TXT format to its proprietary ERA format.

The tool also has a built-in concordancer which allows users to search for and analyse specific error tags, facilitating the identification of patterns and frequencies in learner errors. Additionally, the software can be used to generate reports on the absolute and relative frequency of various error types.

Finally, the tool offers the functionality to create custom exercises tailored to the errors annotated in the texts, enabling educators to address specific learner difficulties more effectively.

4. Case Study 1: Annotating Errors in a Corpus of Learner-Chatbot Interactions

4.1. Overview

The chapter examines the implementation of error annotation using the Louvain Error Tagging Manual Version 2.0 (Granger *et al.*, 2022). Section 4.2 introduces the research setup for the pilot study aimed at refining the taxonomy, followed by a presentation of the three rounds of annotations conducted. The chapter then analyses the distribution of error categories in the third-round annotation at different levels: in the entire sample corpus, by English proficiency level, and across levels (Section 4.3).

4.2. Research Setup for Case Study 1

To evaluate the applicability of the Louvain Error Tagging Manual to the UNITE learner corpus, a pilot study was conducted. The study involved a three-round annotation process on a representative sample of 23 texts from the corpus, selected from the available texts at the time of the study (October-November 2024)—i.e., those with standardized metadata and structural annotation. Moreover, the selection was based on proficiency level and chatbot type to ensure a balanced sample: four texts for each level of proficiency (except C2 level, which only had three usable texts), comprising 12 conversations with Pi.ai and 11 with ChatGPT. Moreover, to capture a wider range of conversational topics, the texts were selected from learners pursuing different academic fields, mainly Social sciences and Education sciences. For details on the sample corpus see Table 15 and Table 16.

Each round of annotation was a direct result of the insights gained from the previous one, providing valuable data on error distribution and providing insights on how to adapt the manual to the conversational context of the UNITE corpus.

Learner profiles = 23	
Gender	▪ Female: 13 ▪ Male: 9 ▪ Non-binary: 1
Area of study	▪ Economics: 1 ▪ Education sciences: 6 ▪ Humanities: 5 ▪ Social sciences: 6 ▪ STEM: 5
Disability	▪ Yes: 3 ▪ No: 20
Self-assessed written English level	▪ A1: 4 ▪ A2: 4 ▪ B1: 4 ▪ B2: 4 ▪ C1: 4 ▪ C2: 3
Certificates	▪ Yes: 8 ▪ No: 15

Table 15 – Learner profiles in sample corpus

Learner EN level	Interactions with Pi.ai	Interactions with ChatGPT
A1	2	2
	4,608 tokens	3,163 tokens
	Learners: 832 tokens	
A2	2	2
	3,360 tokens	8,156 tokens
	Learners: 855 tokens	
B1	2	2
	4,674 tokens	2,717 tokens
	Learners: 1,340 tokens	
B2	2	2
	3,502 tokens	5,839 tokens
	Learners: 2,591 tokens	
C1	2	2
	3,635 tokens	5,890 tokens
	Learners: 2,174 tokens	
C2	1	2
	4,473 tokens	3,918 tokens
	Learners: 1,251 tokens	
Total	12	11
	24,252 tokens	29,683 tokens
	Learners: 9,043 tokens	

Table 16 – Text and tokens distribution in sample corpus

4.2.1. First-Round Annotation

The first round of annotation involved the verbatim application of the Louvain Error Tagging Manual's taxonomy to the selected texts using the UCLEE software (Figure 9). While the Louvain manual was originally designed for annotating written essays in the ICLE corpus, the UNITE corpus contains informal conversational exchanges. Indeed, already during this first phase, the need for adaptations to address specific features of the UNITE corpus soon emerged.

<turn type="student"> FS: i → I GVT: saw → have seen Annihilation and found it a bit underwhelming LSADV: really → actually . FS: but → But yeah, LP: lot of → a lot of food for thought QM: \0 → . </turn>
<turn type="chatbot" who="Pi.ai"> It seems like you've seen a lot of the popular sci-fi movies and shows. Let me suggest something a bit more obscure. Have you heard of "Primer" (2004)? It's a low-budget, mind-bending time travel movie that's become a cult classic. It might not have the big budget effects of Dune or Interstellar, but it makes up for it with its complex plot and interesting ideas about causality and determinism. </turn>
<turn type="student"> GPP: \0 → I GVT: watched → have watched that too, FS: i → I GVT: thought → think GPP: i → it was a bit convoluted, not exactly the time travel story FS: i → I like QM: \0 → , but it was interesting QM: \0 → . </turn>

Figure 9 – Example of error annotation in UCLEE ("Check all" mode)

First, instances of "unwarranted use of mother-tongue words" (Granger *et al.*, 2022: 10) were categorised under the Formal Spelling (<FS>) tag in the original manual. However, in the UNITE corpus these instances can be considered cases of code-switching—defined as "the alternation of two languages within a single discourse, sentence or constituent" (Poplack, 1980: 583). Hence, to better capture their unique nature, a new category, called Code-Switching (<CS>), was introduced. In this phase, this new tag covered both intra-sentential (i.e., switch within a sentence) and inter-sentential code-switching (i.e., switch between sentences) (Poplack, 1980), reflecting a distinction that is often mentioned in the literature.

Second, adjustments were made to the treatment of the Infelicities (<Z>) category. As such, elements like slang, abbreviations, and informal expressions—classified as infelicities in the manual—were excluded from annotation in this study. Their prevalence in casual conversations made them unsuitable as markers of linguistic deviation in this context. Additionally, the use of specific English variants (e.g., American vs. British English) was not tagged as infelicity, contrary to the original manual's specifications. Since participants were not instructed to adhere to any particular variant, marking such usage as infelicities was not necessary for the conversational nature of the data.

Once the annotation process was finalised, a Python script was employed to extract the annotated students turns into TXT files, resulting in a dataset of 9,043 tokens. For subsequent frequency analysis of error distribution, a report on normalised frequencies, calculated per 1,000 tokens (henceforth: ptt), was directly downloaded from the UCLEE software.

The analysis revealed that two macro-categories, Formal errors (79.29 ptt) and Punctuation errors (54.96 ptt), were substantially more prominent than others in the dataset, followed by Grammar errors (39.48 ptt). All remaining categories showed relative frequencies below 15 ptt, underscoring the dominance of the Formal and Punctuation error categories in the dataset.

The results in the UCLEE concordancer suggested that these high frequencies could be influenced by the frequent use of features typical of digitally-mediated communication (DMC)—also known as computer-mediated communication—namely:

- *Uncapitalized first-person singular pronoun*: A common feature in informal digital writing (Tagg, 2012: 14), which was tagged as <FS> and accounted for 25.63% of the Formal errors category.
- *Lowercase letters at the beginning of conversational turns*: Another frequent feature (Lyddy *et al.*, 2014: 547; Yudytska, 2024), tagged as <FS> and representing 24.65% of the Formal errors category.
- *Lack of punctuation marks*: Especially the omission of full stops at the end of conversational turns (Lyddy *et al.*, 2014: 547; Tagg, 2012: 14), which was tagged <QM> and made up 62.17% of the Punctuation errors category.

These features, commonly associated with DMC, reflect informal communication habits typical of online environments and text messages (Tagg, 2012; Yudytska, 2024). This phenomenon is also known as 'textisms', which reflects not only an attempt to emulate features of informal spoken interactions but also to prioritise brevity (Tagg, 2012: 72–73).

Moreover, the absence of full stops in textisms is widely recognised as a characteristic of DMC rather than an actual linguistic error. As noted by Houghton *et al.* (2018), omitting full stops in text messages can often serve a rhetorical function rather than a strictly grammatical one. Indeed, the presence or absence of a full stop can also convey pragmatic nuances, with a period often interpreted as abrupt.

The prominence of these features in the data raised concerns that they might disproportionately influence the overall error distribution results. To attempt to correct this imbalance, a second-round annotation was conducted, aiming to evaluate the impact of excluding such features on the error distribution analysis and to eventually refine their categorisation.

4.2.2. Second-Round Annotation

The second-round annotation focused on refining the error distribution by eliminating the previously identified DMC features from the annotated turns. To accelerate the process, find-and-replace operations were performed in Notepad++, allowing for the rapid removal of these instances. Subsequently, a new report on relative frequencies was generated and downloaded from the UCLEE software for further analysis. The updated frequency analysis revealed a more balanced error distribution across the macro-categories. The previously dominant Formal (39.37 ptt) and Punctuation (21.12 ptt) error categories were now levelled out, with the Grammar error category (39.48 ptt) emerging as the most prevalent.

These findings suggested that the DMC features were influencing the error distribution. Since these features appeared to be prevalent in the UNITE corpus, it was determined that they required a different approach, and that the best way to track them was putting them in a separate category. As for missing full stops at the end of conversational turns, given that they are a very common feature of textisms and often perceived as abrupt rather than incorrect, the decision was to continue not to consider them as errors. These insights led to a third-round annotation and an adaptation of the error annotation system to better capture the unique characteristics of the corpus.

4.2.3. Third-Round Annotation

Based on the insights gained from the previous two rounds of annotation, a refined error taxonomy was implemented on the annotated turns and designated as the final version. This revised taxonomy (see Appendix B for the complete list) introduced two new error categories to better address the specific features of the UNITE corpus:

1. **Digitally-Mediated Communication:** This category was subdivided into two subcategories:
   - *Capitalization-related issues* (<DMCC>): For non-standard use of capitalization, such as uncapitalized first-person singular pronoun or the first word of a sentence.
   - *Abbreviations* (<DMCA>): For instances of common abbreviations used in digital contexts (e.g., "OMG", "lol", etc.) (Verheijen, 2015).
2. **Code-Switching:** Differently from the first-round annotation (see Section 4.2.1), intra-sentential and inter-sentential code-switching were differentiated into two distinct subcategories, each with its own tag, <CS-INTRA> and <CS-INTER> respectively .

Furthermore, word-coinage and calques, previously categorised under the Formal Spelling (<FS>) tag in the Louvain Error Tagging Manual, were reclassified into a new subcategory, <LWCO>, under the Lexis macro-category. This reclassification aligns with the framework proposed for intercomprehension by Cervini and Paone (2024: 503), where calques are classified as pertaining to the lexical level. This new subcategory also includes instances of incorrect or invented proper names.

These adjustments provide a more precise categorisation of the linguistic features of the UNITE corpus, resulting in nine macro-categories and a total of 59 tags. Following the implementation of the refined taxonomy, a final analysis of error distribution was conducted. The results, which provide an in-depth view of error patterns in the corpus, will be discussed in greater detail in the following sections.

4.3. Error Distribution Analysis

The analysis of error distribution is based on the third-round annotation (see Section 4.2.3), where the corpus texts were annotated using the refined and adapted error taxonomy based on the Louvain Error Tagging Manual. As in the first- and second-round annotations, normalised frequency per 1,000 tokens was used as the metric of comparison.

4.3.1. The Distribution of Error Macro-Categories

Looking at Figure 10, which refers to the entire sample annotated corpus described in Section 4.2, it is possible to compare the frequency of different error macro-categories. The most prominent category is Digitally-Mediated Communication (DMC) with a frequency of 45.67 ptt, followed by Grammar (39.48 ptt) and Form (33.62 ptt).

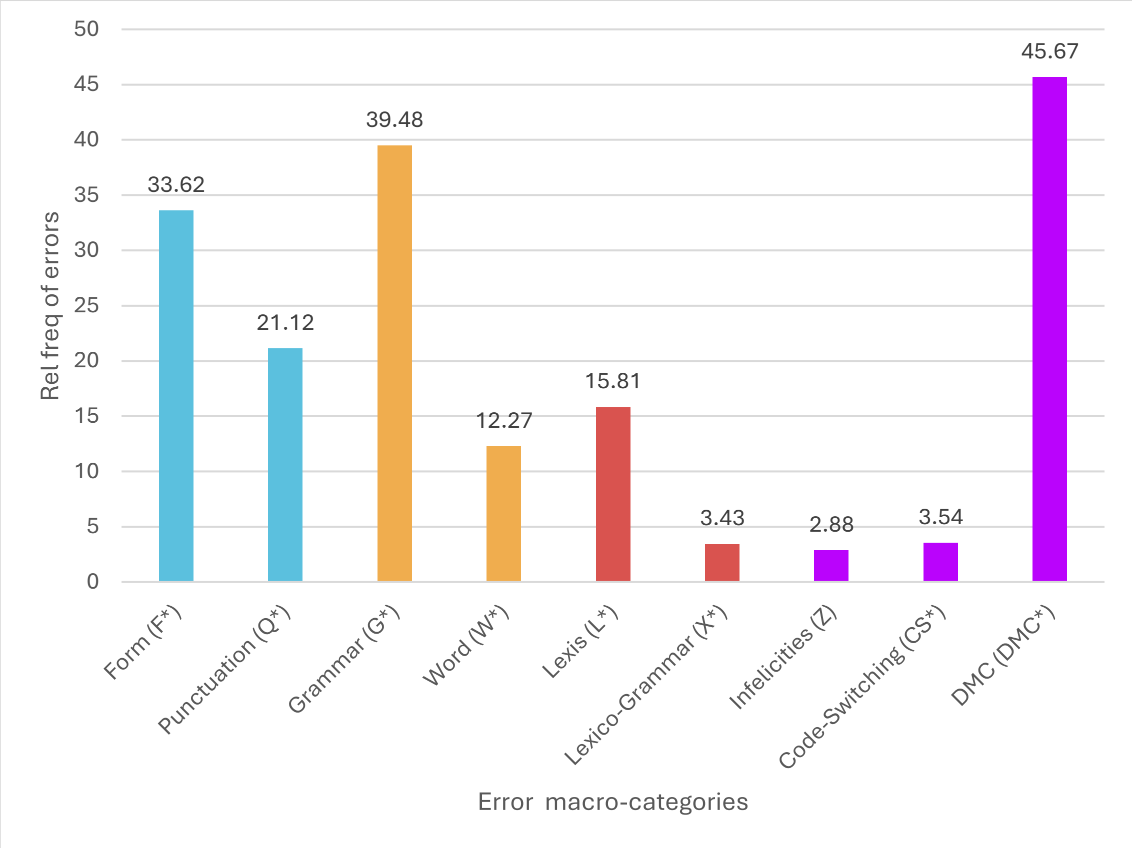

Figure 10 – Distribution of error macro-categories (per 1,000 tokens)

The high frequency of DMC instances supports the hypothesis formulated during the first- and second-round annotations (see Section 4.2.1 and Section 4.2.2), suggesting that these features—strongly connected with the conversational nature of the interactions in the UNITE corpus—were strongly influencing the frequency of errors originally categorised under Form. Furthermore, compared to the second-round annotation, the Form category shows a slight additional decrease from its previous frequency of 39.37 ptt, which can be attributed to two additional adjustments:

1. Uncapitalized initial letters after full stops were now tagged as Digitally-Mediated Communication Capitalization (<DMCC>) instead of Form Spelling (<FS>).
2. Lexical coinages and calques were reclassified under the newly introduced subcategory Lexis Word Coinages (<LWCO>), rather than Form Spelling (<FS>), leading to a slight increase in the Lexis category (15.81 ptt, compared to the previous value of 14.15 ptt).

Another observation, already confirmed during the second-round annotation, concerns the Punctuation category. Excluding missing full stops at the end of conversational turns from the error count led to a more balanced error distribution and a more accurate comparison with errors from the other categories. Indeed, the results of the third-round annotation place Punctuation (21.12 ptt) as the fourth most frequent error category, whereas in the first- and second-round annotations, it ranked second and third, respectively.

Beyond the four dominant categories, Word errors (12.27 ptt) emerge as the fifth most frequent, showing that issues related to word order, as well as redundant and missing words, are also relevant elements within the corpus. This category is then followed by Code-Switching (3.54 ptt), Lexico-Grammar (3.43 ptt), and Infelicities (2.88 ptt), which represent the least frequent error types. From this, it is possible to notice that Code-Switching instances, even if not as prominent as other categories, are slightly higher than Infelicities and Lexico-Grammar errors.

Furthermore, it is also possible to analyse the data on a higher level of granularity by grouping the macro-categories under broader linguistic areas:

- Orthographic and morphological errors: Form and Punctuation.
- Syntactic and morphosyntactic errors: Grammar and Word.
- Lexical and lexico-grammatical errors: Lexis and Lexico-Grammar.
- Pragmatic and stylistic errors: Infelicities, Code-Switching, and DMC.

This grouping is visually represented in Figure 10, where the same colour was assigned to macro-categories belonging to the same linguistic areas. Hence, by taking into account the total normalised frequency value of each linguistic area, orthographic and morphological errors emerge as the most prominent (54.74 ptt), followed very closely by pragmatic and stylistic errors (52.09 ptt), and syntactic and morphosyntactic errors (51.75 ptt). In contrast, lexical and lexico-grammatical errors are the least frequent (19.24 ptt).

Finally, an additional analysis can be done by comparing the least and most serious errors, assessing their severity in terms of their impact on text comprehension. In this regard, orthographic, morphological, pragmatic and stylistic errors tend to affect less comprehensibility, and in this case, they account for a frequency value of 106.83 ptt. On the contrary, syntactic, morphosyntactic, lexical, and lexico-grammatical errors can be considered the most serious, yet in this case, they seem to be less frequent (70.99 ptt) compared to the least serious errors.

4.3.2. Error Frequency by English Level

Going more in detail, Figure 11 shows the frequency of errors for each English proficiency level. As one would expect, the highest frequency of errors is registered at lower proficiency levels, with a slightly higher error rate for A2 level (394.15 ptt) compared to A1 level (353.37 ptt).

Error frequencies decrease substantially at intermediate and advanced levels, with B1 and B2 having respectively a frequency of 197.76 ptt and 115.01 ptt. Even if this confirms an increase in linguistic accuracy as proficiency level rises, it is also worth noting that the decline continues only until C1 level (95.68 ptt). C2 learners, despite their advanced proficiency, present a higher error frequency (164.67 ptt) compared to B2 and C1 levels.

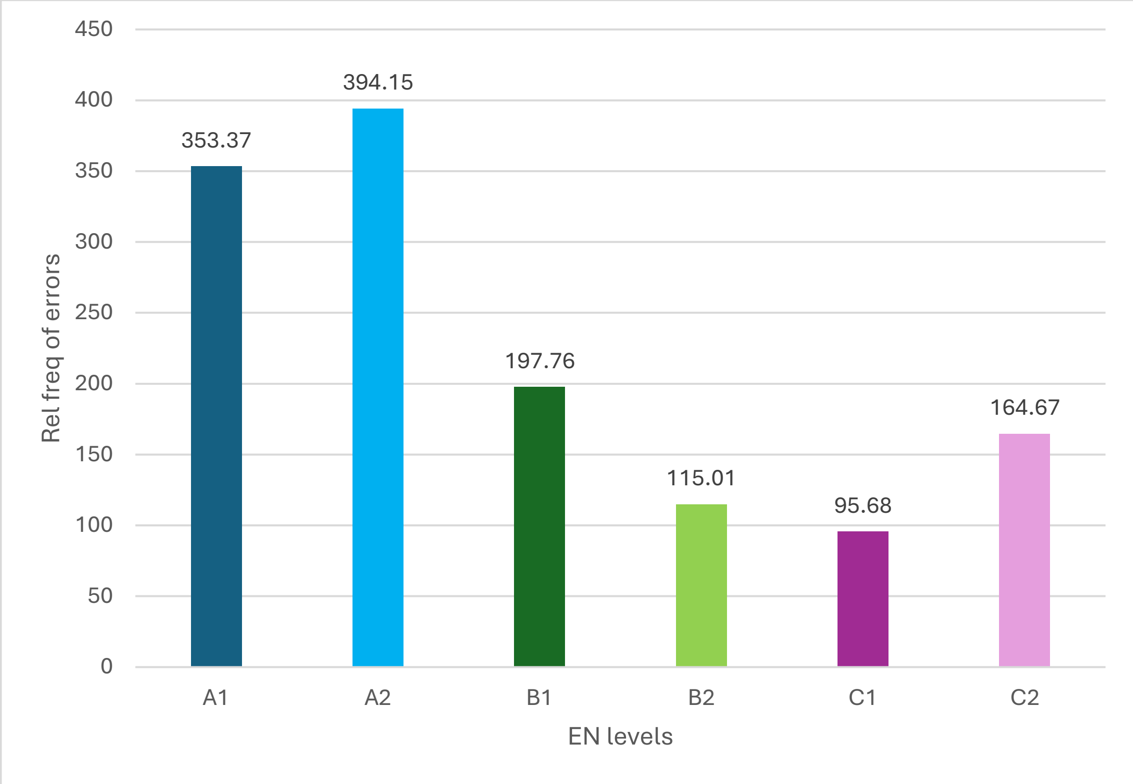

Figure 11 – Error frequency by EN level (per 1,000 tokens)

4.3.3. Error Macro-Categories Distribution Across English Levels

After analysing the general trends, it is also valuable to examine how the frequency of each macro-category of errors changes by proficiency level. Table 17 provides a detailed comparison of the actual frequency values of each macro-category at each proficiency level.

EN level	Form (F*)	Punctuation (Q*)	Grammar (G*)	Word (W*)	Lexis (L*)	Lexico-Grammar (X*)	Infelicities (Z)	Code-Switching (CS*)	DMC (DMC*)
A1	64.9	18.03	90.14	36.06	48.08	6.01	7.21	6.01	76.92
A2	71.35	36.26	92.4	32.75	37.43	9.36	2.34	12.87	99.42
B1	32.84	23.13	36.57	14.18	11.19	6.72	2.24	3.73	67.16
B2	25.47	16.98	24.7	4.25	8.1	1.93	3.86	2.7	27.02
C1	17.02	26.22	26.68	1.38	8.74	0.46	0.92	0	14.26
C2	33.57	10.39	25.58	15.99	12.79	2.4	2.4	3.2	58.35

Table 17 – Frequency (ptt) of each error macro-category for each English level

Figure 12 visually presents the corresponding patterns. Looking at the general trend, as one would expect, the highest error frequency is registered at the A1 or A2 levels across all categories. Moreover, all categories seem to align to some extent with the global trend described in Section 4.3.2, showing (more or less) substantial peaks at A2 and/or C2 level.

By analysing the data using the broader linguistic areas outlined in Section 4.3.1, it is possible to examine the different patterns in greater detail. For the sake of visual clarity, the same colour-coding scheme as in Section 4.3.1 was applied, with each sub-level of granularity (i.e., each category within each linguistic area) further differentiated using different line types. As for the categories under orthographic and morphological errors, both Form and Punctuation present peaks at A2 level and again at higher proficiency levels—Punctuation at C1 and Form at C2. Regarding syntactic and morphosyntactic errors, the Word category follows a decreasing trend but presents a notable peak at C2 level, while Grammar shows slight increases at A2 and C1. Actually, this makes Grammar the category that aligns most consistently with general expectations, as it generally demonstrates a clear reduction in errors as proficiency increases. For lexical and lexico-grammatical errors, the Lexis category follows a decreasing curve until B2, after which it begins to rise slightly again up to C2 level. Instead, the Lexico-Grammar category remains relatively stable across all levels, except for slight peaks at A2 and C2, perfectly aligning with the global trend in Section 4.3.2. Finally, as for pragmatic and stylistic errors, the Infelicities curve shows peaks at B2 and C2 compared to lower proficiency levels. Instead, both Code-Switching and DMC follow the global trend—just like Form and Lexico-Grammar—presenting peaks at A2 and C2. Furthermore, DMC is also the category with the most substantial peaks at these two levels compared to all others.

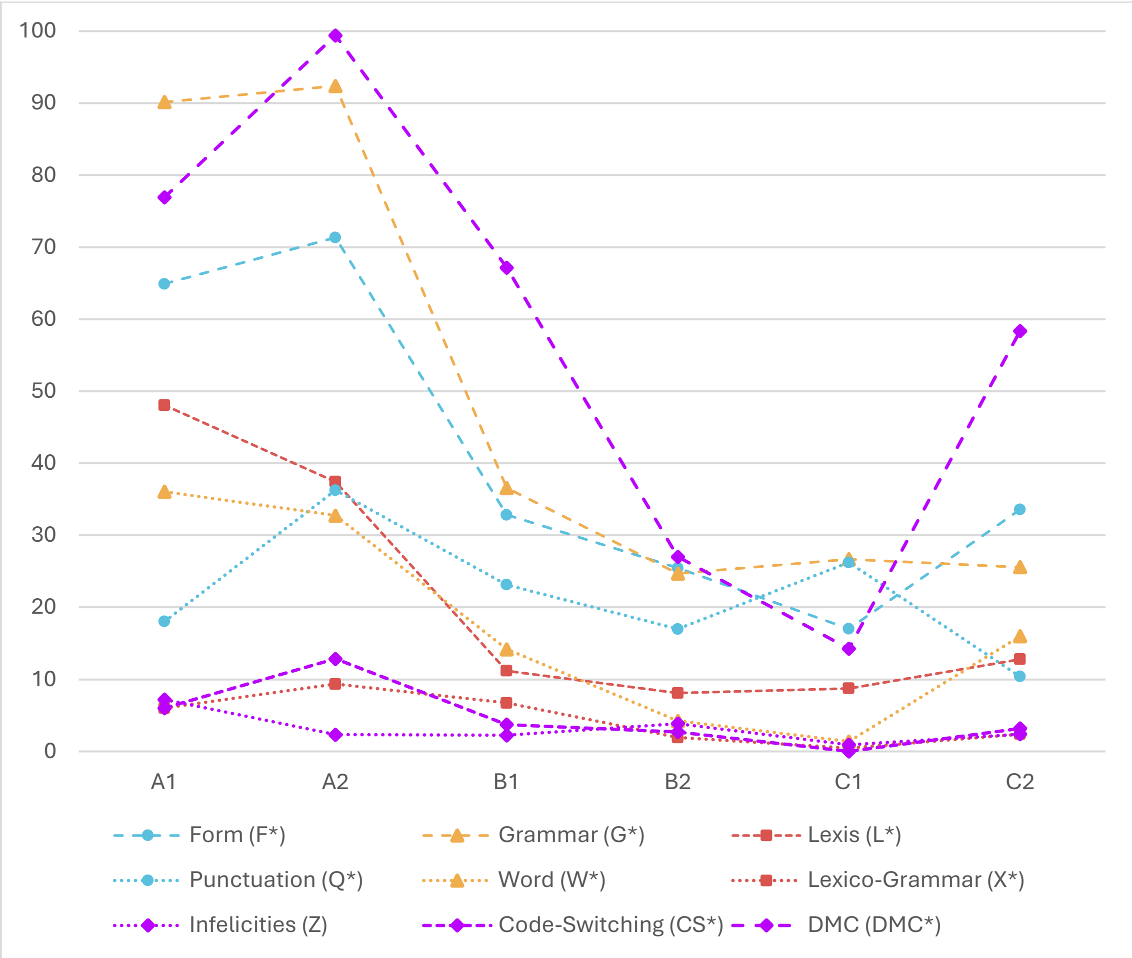

Figure 12 – Error macro-categories distribution across EN levels (per 1,000 tokens)

5. Case Study 2: Using ChatGPT for Error Annotation Tasks

5.1. Overview

This chapter addresses the use of ChatGPT as a proof-of-concept experiment for automated annotation tasks. After explaining the research setup for the study, Section 5.2 focuses on the adaptation of the error tagging manual, the creation of a custom GPT for error annotation, and its testing through two different approaches: step-by-step and full-text annotation. Finally, the chapter illustrates the results on the model's output accuracy, presenting data for both approaches (Section 5.3).

5.2. Research Setup for Case Study 2

As also highlighted in the limited but emerging research on the use of LLMs for corpus annotation, the manual annotation process can be both very time-consuming and susceptible to errors (Imamovic *et al.*, 2024: 112), often caused by annotators' cognitive fatigue (Yu *et al.*, 2024: 6). These challenges become particularly significant when dealing with large datasets, such as the considerable number of conversations in the UNITE corpus. For this reason, as a continuation of the pilot study on error annotation, the next step was to explore the potential of using an LLM to perform a preliminary annotation of errors in students' conversational turns, with the aim of accelerating and partially automating the process.

For this proof-of-concept experiment, only ChatGPT was chosen as chatbot, as it supports file attachments and multiple chat modes—basic chat, Project, and custom GPT (see Section 5.2.2)—compared to Pi.ai which, at the time of writing, lacks file attachments and is limited to basic chat mode. The possibility of attaching files was an essential requirement for providing the chatbot with the error annotation manual (see Section 5.2.1), while multiple chat modes enabled broader testing to determine the most effective approach for accurate annotation.

To undertake this task effectively, several key factors had to be taken into consideration. These included the content and structure of the prompt provided to the chatbot, ensuring that it could accurately interpret and apply the tagging conventions, as well as the optimal format for presenting both the error tagging manual and the texts for annotation.

5.2.1. Adapting the Error Tagging Manual

The first step of this experiment consisted in adapting the original Error Tagging Manual to incorporate the refined taxonomy developed for the third-round annotation (see Section 4.2.3). The chosen format for rewriting and adapting the manual was the Markdown (MD) format (Figure 13), which offers several advantages:

- It is easily transferable and platform independent, i.e., it can be opened using any application (Cone, 2024).
- It can be used for web-based applications (Cone, 2024).
- ChatGPT supports and interprets Markdown syntax with ease (OpenAI, 2024b).

````
<a name="3-error-categories"></a>
## 3. Error Categories

<a name="31-digitally-mediated-communication"></a>
### 3.1. Digitally-Mediated Communication

<a name="capitalization-dmcc"></a>
#### Capitalization: `<DMCC>`

Use `<DMCC>` exclusively for **capitalization-related errors that fall into these categories**:

- Uncapitalized first person singular pronoun "I".
- Uncapitalized word at the beginning of a sentence (after full stop, exclamation or question mark).
  - **Example:**
    ```xml
    As you think it's the best. <DMCC corr="We">we</DMCC> trust your taste
    ```
- Uncapitalized word at at the beginning of the conversational turn.
  - **Example:**
    ```xml
    <DMCC corr="Hi">hi</DMCC> there, thank you for asking me.
    ```

**Tip:** If the uncapitalized word is a proper noun or contains other spelling issues (beyond capitalization), use [`<FS>`](#spelling-fs) instead.
````

Figure 13 – Snippet of error annotation manual in Markdown format

In addition to incorporating the new error categories mentioned in Section 4.2.3 and revisions to the Infelicities category in Section 4.2.1, the adapted manual features some significant changes compared to the original manual:

1. *Corpus-based examples:* It contains examples from the UNITE corpus[17].
2. *Exclusion of tagging principles:* It does not contain the tagging principles section, incorporating these guidelines directly into ChatGPT's instructions (see Section 5.2.2 and Appendix C).

[17] The examples were selected from 12 of the 23 annotated texts to avoid influencing ChatGPT's output when annotating the remaining sample dataset. For each error tag, only the error relevant for that tag was retained, even if the original example included multiple errors. When no suitable example was available, an example from the original manual was used.

3. *Extended use cases of tags:* The descriptions and use cases for certain categories were expanded to reflect common issues identified in the UNITE corpus (e.g., now the <GVM> (Grammar Verb Morphology) tag explicitly covers the use of non-*ing* verb forms in continuous tenses, unlike the original manual which does not explicitly address this issue, mentioning only errors related to incorrect formations of past verb tenses or the third person singular).
4. *Use of chatbot-friendly language:* Explanations and exceptions were revised with in mind the best practices for optimising chatbot's ability to process and interpret instructions effectively (cf. OpenAI, 2024b). This included making descriptions more concise, replacing warnings with practical "tips", employing imperatives, avoiding complex negative structures where possible, and using both correct and incorrect examples—similar to few-shot prompting[18], i.e., a technique used in prompt engineering where the chatbot is provided with several examples in order to improve its output (Dang *et al.*, 2022).
5. *Annexes with tables:* An annexes section with tables was added to improve comprehension of complex or "fuzzy" areas of annotation (e.g., specifications on when to use <FS> instead of <DMCC>, or determining proper tag placement), particularly where the chatbot tended to hallucinate, i.e., to generate a plausible but incorrect or nonsensical answer (Athaluri *et al.*, 2023).

The final structure of the manual—comprising an introduction, a tag reference table, a detailed list of error categories with explanations and examples, and an annexes section with tables—was refined through iterative testing and evaluation of ChatGPT's outputs, with the use of tables proving especially effective in improving the chatbot's understanding of this complex tagging process.

While the primary goal was to optimise the manual for ChatGPT, the revised structure and content also offer significant benefits for human annotators, providing clear, concise, and user-friendly guidelines for error annotation.

[18] Depending on the number of examples given to the chatbot, it is also possible to talk about zero-shot prompting (zero examples) and one-shot prompting (one example) (Dang *et al.*, 2022).

5.2.2. Creating a Custom GPT

ChatGPT-4o[19] was the version of ChatGPT selected for the annotation task due to its advanced capabilities as one of OpenAI's latest model, offering enhanced data analysis skills and the ability to process uploaded reference files (OpenAI, 2024a). Determining the best approach for the annotation task involved several attempts and required careful consideration of the model's functionalities and limitations.

The initial attempts focused on exploring the basic Chat and Project modes. In the former, the chatbot was provided with the manual alone, as well as in combination with sample annotated texts, to assess its ability to follow instructions and apply error tags. Similarly, the Project mode—which allows users to organise chats, files, and custom instructions into a single workspace (OpenAI, 2024c)—was tested using the same approach: the manual was uploaded as project file both separately and in combination with sample annotated texts in two different rounds. However, these approaches revealed some limitations: the chatbot frequently lost track of the instructions and struggled to consistently refer to the error taxonomy.

These challenges led to the exploration of the custom GPT feature which enables users to combine inner instructions with knowledge files. This functionality allows for customization of the model's behaviour, making it possible to tailor ChatGPT for specific tasks or topics (OpenAI, 2025). This translated in higher possibilities that the model could consistently refer to the error tagging manual and follow the specific guidelines provided for the annotation task.

Another critical aspect to be tested was determining the best format and content for the annotation files. After several attempts, it became clear that the TXT format was the most effective for uploading files, since this format is easily readable by ChatGPT. Moreover, it was decided to include only students' conversational turns in the files. This approach streamlined the annotation process by ensuring that the chatbot could directly focus only on students' turns, eliminating the need to instruct the model to ignore chatbot's turns.

---

[19] This study was conducted in January 2025. To access the full capabilities and features of ChatGPT-4o, it was necessary to use the subscription-based ChatGPT Plus plan. While this requirement may limit the free replicability of the experiment, it is worth noting that future advancements may lead to this model becoming fully accessible as more advanced models are released.

After evaluating all options, a custom GPT, named *Learner Corpora Annotator* (Figure 14), was chosen as the final approach for several reasons. First, it allowed the manual to be uploaded as a knowledge base, ensuring the chatbot consistently referred to it when performing annotations. Second, it provided a structured environment in which the model's behaviour could be fine-tuned through detailed inner instructions, reducing deviations and errors observed in earlier tests. These features collectively enhanced the reliability and precision of the annotation process: chat mode, with only the annotation manual provided, achieved an accuracy of 17.02%, while with both the manual and sample annotated texts, it scored 12.77%. The latter value was also achieved by Project mode for both configurations. In contrast, the custom GPT achieved an accuracy approximately three times higher for the same annotated text. For a comprehensive analysis of accuracy results of the custom GPT's accuracy results and the methodology used for calculation, see Section 5.3.

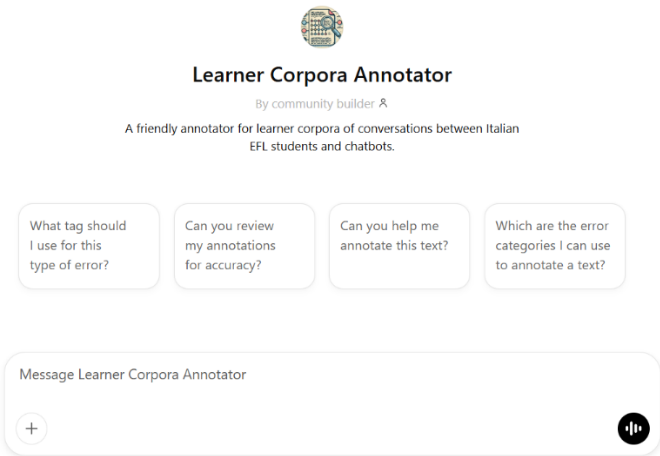

Figure 14 – GPT Learner Corpora Annotator interface

The creation of effective instructions for the custom GPT was a pivotal aspect of this experiment. The final instructions resulted from an iterative process, involving repeated testing of ChatGPT's outputs. Following recommendations from OpenAI website (cf. OpenAI, 2024b), the instructions were carefully structured to enhance the model's comprehension and performance. Key features of the instructions included:

- *Markdown syntax*: The instructions were formatted in Markdown, making them easy to read and process.
- *Trigger/instruction pairs*: Use of clearly defined triggers tied to specific actions.

```
**Trigger:**
The user asks to annotate a text file.

**Instructions:**
1. **Preparation**
  - Always access the provided `annotation_manual.md` in the knowledge section and revise
  error tags to ensure familiarity.
  - Ask the user if they want you to annotate:
    - Only the first two student turns as a checkpoint.
    - A specific section of the file.
    - The entire file in one go (with an option for the user to review). Default to the first two turns
    if unclear.
  - If the user gives your further instructions, include them in the annotation process.
[...]
```

- *Step-by-step instructions*: Multi-step tasks were broken down into simpler, manageable steps to implement a chain-of-thought approach.

```
2. **Annotation**
  - For each student's turn in the agreed portion of the file break down the process into
  manageable steps:
    - Apply the guidelines provided in the in next section of these instructions (**"Tagging
    Guidelines"**).
    - Analyze the content and identify potential issues.
    - Choose the right tags.
    - Verify whether each identified issue constitutes an actual error as per the
    `annotation_manual.md`.
[...]
```

- *Few-shot prompting*: Providing examples of inputs and correct outputs for clarity.

```
Only tag capitalization errors if they clearly violate the rules in annotation_manual.md.
  - **Example of Original:** `Hi how are you? `
    **Example of Output:** `Hi how are you? ` (leave untagged)
  - **Example of Original:** `hi how are you? `
    **Example of Output:** `<DMCC corr="Hi">hi</DMCC> how are you? `
```

- *Reference to knowledge base*: Instructions explicitly referenced the error tagging manual's filename and asked the chatbot to consistently consult it.

```
Always access the provided `annotation_manual.md` in the knowledge section and revise error
tags to ensure familiarity.
```

Additional insights for refining the content of the instructions were also taken from previous experiments, such as Yu *et al.*'s work (2024: 12), which emphasised the importance of prompt's formal layout, textual conciseness, and explicitness. Special attention was also given to avoiding as much as possible common hallucinations observed during testing. To address this, the instructions were organised into the following sections:

1. **Context:** Overview of the GPT's role and behaviour, as well as of the content of the knowledge base.
2. **Annotation Process:** Detailed steps for preparing, annotating, reviewing, and finalising the annotations.
3. **Tagging Guidelines:** Clear and concise tagging principles with examples for handling the annotation task.
4. **Error Explanation Requests:** Instructions for dealing with clarification requests from the user.

The final version of the instructions is provided in Appendix C for reference. As for the knowledge base, only the error tagging manual was included. Initial attempts to incorporate sample annotated texts alongside the manual were abandoned, as these seemed to confuse the model, likely due to token processing constraints. In addition to the knowledge base, several advanced settings were enabled to enhance the model's functionality, including the enabling of Web Search, Canvas, and Code Interpreter & Data Analysis (necessary for interpreting the uploaded files).

While the primary goal of this configuration was to facilitate error annotation tasks (described in detail in Section 5.2.3), the custom GPT also proved useful for other tasks. For example, it was effective in answering clarifications about the tagging manual, such as questions on tag placement and usage. Furthermore, the custom GPT holds potential for reviewing annotations, even though this function remains to be fully tested.

5.2.3. Testing the Custom GPT

Once the *Learner Corpora Annotator* GPT was fully configured and ready for use, annotation tasks were implemented to evaluate its performance. When prompted to annotate a text, the GPT offered two options: annotating two conversational turns (or another agreed portion of the file) or annotating the entire text in one go (Figure 15).

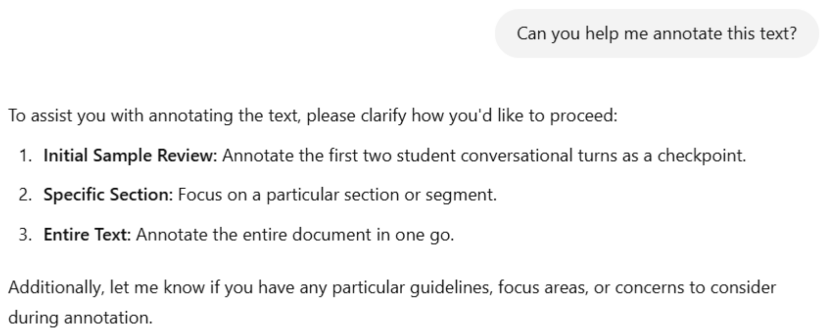

Figure 15 – Starting the annotation task

Both approaches were tested using a carefully designed prompt that was the result of a progressive trial-and-error approach. The prompt was refined to ensure that the GPT adhered to the required tagging requirements and produced the most accurate output possible. The instructions contained in the prompt were aimed at:

- *Specifying the output format*: The prompt contained indications to use XML format to show the annotated output, using examples from the sample annotated texts to guide the model (few-shot prompting). Each example showed both the original and annotated versions of a conversational turn[20].

[20] Keeping track of the original versions was essential, as the chatbot showed a tendency to hallucinate by capitalizing the beginning of conversational turns or adding a full stop at the end, even when these features were not present in the original text.

````
Annotate the entire document and show it in XML format.

Example:
```xml
1. Original: I woul like to tolck about my travel in Malta
Annotated: I <FS corr="would">woul</FS> like to <FS corr="talk">tolck</FS> about my <LSN corr="trip">travel</LSN> <LSPR corr="to">in</LSPR> Malta

2. Original: i would like only specking about you
Annotated: <DMCC corr="I">i</DMCC> <XVCO corr="would like only to speak">would like only <WM corr="to">\0</WM> <FS corr="speaking">specking</FS></XVCO> about you
```
````

- *Reinforcing tagging guidelines*: The prompt asked the chatbot to pay attention to important tagging requirements (also outlined in the manual and inner instructions). The inclusion of these elements was the result of challenges observed during initial tests, e.g., issues with tag placement.

```
While annotating pay attention to:
- Choose the tag on the basis of the incorrect element.
- Where to correctly place the tag.
- How to tag capitalization issues.
[...]
```

- *Encouraging the GPT to be thorough*: Following OpenAI's recommendations (cf. OpenAI, 2024b), the prompt included the sentence 'Take your time to analyse each sentence thoroughly', aimed at encouraging the model to prioritise attention to detail.

The first annotation task involved a step-by-step process, where the GPT was instructed to annotate a range of three to five conversational turns at a time. Each request used the same prompt to ensure consistency. This method allowed for immediate feedback on its output, especially where the GPT violated tagging rules, for example, on tag placement (Figure 16).

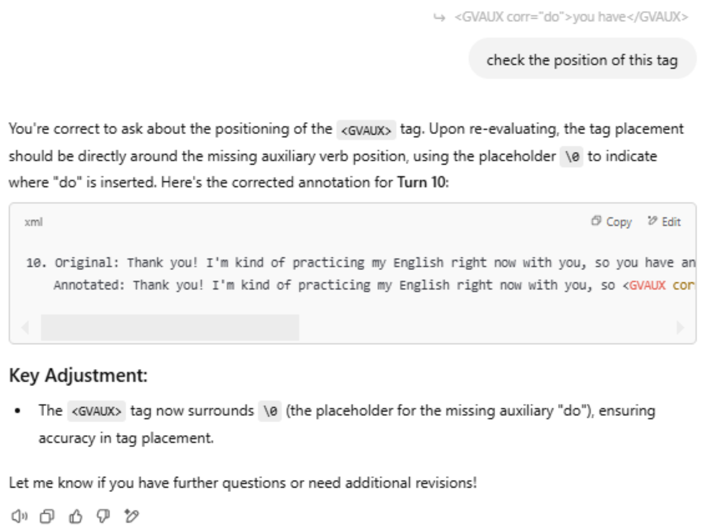

Figure 16 – Example of feedback during step-by-step annotation process

After completing the annotation of each turn, the GPT was prompted to save the annotated turns in a TXT file. Specific requirements were provided regarding:

- Formatting and encoding

```
I would like to save the annotation in TXT format using UTF-8 encoding:
- Only include the annotated version of each turn, even those where no correction was made.
- Separate each turn using `\n\n`.
```

- The use of specific characters (e.g., “<” and “>” for tags rather than their encoded equivalents “<” and “&gt”)

```
- Use the literal string `\0` (backslash zero) for missing words or deletions, not the NUL character
(`\x00`).
- Use `<` and `>` for tags, not `&lt;` and `&gt;`.
```

- File naming

```
Append `_gpt` to the original filename (e.g., `example.txt` → `example_gpt.txt`).
```

For the full prompts, see Appendix D. This approach proved effective in ensuring that the GPT adhered to the main tagging rules and produced outputs of acceptable quality. However, while the step-by-step method provided a high level of control, it also required considerable time and effort, making it less practical for annotating large datasets.

To address the time constraints of the step-by-step approach, the second strategy involved annotating entire texts in a single request. For this test, the GPT was asked to process one text at a time (for a total of three texts), using the same prompts and guidelines as in the step-by-step method. By shifting to full-text annotation, the concern was that the lower level of control could increase the likelihood of errors and inconsistencies in the outputs. However, not only did the process become significantly faster, but accuracy did not seem to be considerably affected, possibly suggesting that the GPT's performance may be more influenced by the difficulty of the text to be annotated rather than by the annotation method itself. The results of these annotation tasks, including an analysis of the accuracy and reliability of the GPT's output, will be presented and in the next section.

5.3. Evaluation of Accuracy: ChatGPT vs. Human Annotation

As described in Section 5.2, the second pilot study served as a proof-of-concept experiment, focusing on testing the reliability of the custom GPT *Learner Corpora Annotator* in identifying and annotating errors in texts from the UNITE corpus. As mentioned in Section 5.2.3, a first attempt of annotation was made using a step-by-step approach on a single text, followed only later by the full-text annotation of three additional texts. This section will provide a detailed analysis of the custom GPT's annotation accuracy in comparison to a human annotation, used as the gold standard. Figure 17 offers a visual example of the comparison between the GPT's annotations and those of the human annotator for the text processed using the step-by-step approach. Already at first glance, it is possible to see that the chatbot's annotations are quite accurate, but fewer in number compared to the human annotations. This is confirmed by the more detailed analysis in Table 18 and Table 19.

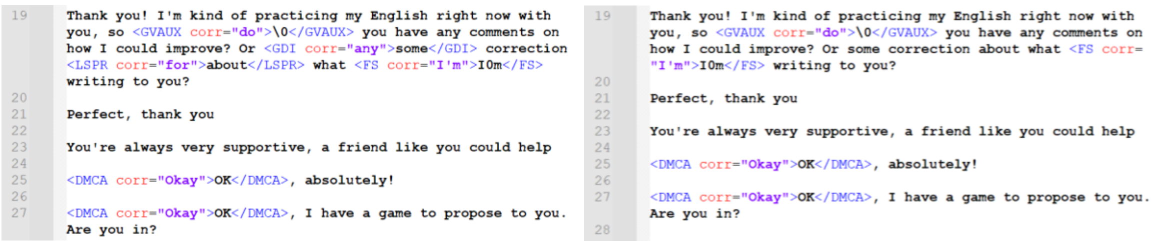

Figure 17 – Human annotation (on the left) vs. GPT's annotation (on the right)

In order to assess the GPT's accuracy, a Python script was used to extract the annotations and categorise them in four different groups:

- *Correct (True Positives – TP)*: Annotations that perfectly match the gold standard.
- *Partially Correct (PC)*: Annotations with at least two matching elements (correction + incorrect text, tag + incorrect text, tag + correction), as well as cases where the incorrect text matches, but the tag and the correction differ.
    - Examples:

        1. correction + incorrect text

        HUMAN ANNOTATOR: <GWC corr="**development**">***developing***</GWC>

        GPT: <*FS* ="**development**">***developing***</FS>

        2. tag + incorrect text

        HUMAN ANNOTATOR: <**FS** corr="strongly">***stregIty***</**FS**>

        GPT: <**FS** corr="strictly">***stregIty***</**FS**>

        3. tag + correction

        HUMAN ANNOTATOR: <**DMCC** corr="**I**">*i*</**DMCC**>'m

        GPT: <**DMCC** corr="**I**">*i'm*</**DMCC**>

        4. same incorrect text but different tag and correction

        HUMAN ANNOTATOR: <GNC corr="constructors'">***constructors***</GNC>

        GPT: <FS corr="constructors">***constructors***</FS>

- *Missed (False Negatives – FN)*: Annotations present in the gold standard but missing in the chatbot's output.
- *Incorrect (False Positives – FP)*: False positives generated by the chatbot, as well as all other incorrect instances not included in the previous categories.
    - Example:

        GPT: <DMCC corr="Well">Well</DMCC>

The accuracy of the GPT's output was calculated based on the standard accuracy formula, which measures the proportion of correctly classified instances (true positives + true negatives) out of the total instances (true positives + true negatives + false positives + false negatives):

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

However, in this study, the formula was adapted to mirror the available data types. First, since human annotations were considered as the gold standard, there were no cases in which both the human and the chatbot did not annotate a given instance (true negatives). Second, as mentioned above, instances of partially correct annotations were treated as a separate category rather than classifying them as strictly correct or incorrect. To account for these adjustments, true negatives were excluded and partially correct annotations added to the total of the total instances. Hence, the formula was modified as follows:

$$\text{Accuracy} = \frac{Correct\ instances\ (TP)}{Total\ instances} = \frac{TP}{TP + PC + FP + FN}$$

The following tables (Table 18 and Table 19) report both the absolute counts and corresponding percentage values for each annotation category. The percentage values of partially correct (PC), incorrect (FP), and missed annotations (FN) were calculated using the same formula, with the respective category taken as the numerator.

Table 18 illustrates the results of the text—produced by a B2-level student—annotated using the step-by-step approach. As could already be grasped by the visual example in Figure 17, the number of GPT's annotations (19) for this text is much lower compared to the human ones (47). Among these, the GPT scored a percentage of 33.33% correct annotations, with two partially correct annotations (4.17%). Of these, one was a case of same CORRECTION + INCORRECT TEXT, while the other was a case of same incorrect text but different tag and correction, meaning that these partially correct annotations did not violate tag placement rules. Indeed, as also mentioned in Section 5.2.3, the step-by-step approach made it possible to guide the chatbot whenever it made a mistake of this nature.

Info		Incorrect (FP)	Missed (FN)	Partially correct (PC)	Correct (TP)	Total
Text (B2 level)	TEXT TOKENS:643 HUMAN ANNOTATIONS: 47 GPT'S ANNOTATIONS: 19	1	29	2	16	48
		2.08%	60.42%	4.17%	33.33%	100%

Table 18 – Accuracy results for step-by-step annotation

Moving to the full-text approach, Table 19 shows the results for the three texts—produced by A1, B2, and C1 students respectively—employed to test this method. Also in this case, the trend of the GPT inserting fewer annotation remains consistent, with only 144 annotations across the three texts, compared to the 230 inserted by the human annotator. Looking at the GPT's overall performance, it achieved an accuracy of 38.25% correct annotations, with 27 partially correct annotations (10.75%). These included all four possible cases: same CORRECTION + INCORRECT TEXT (14 occurrences), same incorrect text but different tag and correction (6 occurrences), same TAG + INCORRECT TEXT (4 occurrences), and same TAG + CORRECTION (3 occurrences). This suggests that even when the annotations do not always perfectly match the gold standard, in most cases the GPT respects tag placement rules and detects the error. However, it should also be considered that all discrepancies found in the annotation of incorrect text segments are due to the chatbot's inclusion of an extended text segment (see example 3 earlier in this section). This means that even if it accurately spots the error, in this case it violates the tag placement rules.

Furthermore, surprisingly, the step-by-step annotation scored a lower accuracy percentage than the full-text approach. Nonetheless, since the two approaches were tested on different numbers of texts, further testing would be needed to confirm this data.

	Info	Incorrect (FP)	Missed (FN)	Partially correct (PC)	Correct (TP)	Total
Text 1 (A1 level)	TEXT TOKENS: 214 HUMAN ANNOTATIONS: 65 GPT'S ANNOTATIONS: 34	4	35	9	21	69
		5.80%	50.73%	13.04%	30.43%	100%
Text 2 (B2 level)	TEXT TOKENS: 537 HUMAN ANNOTATIONS: 96 GPT'S ANNOTATIONS: 59	5	42	11	43	101
		4.95%	41.58%	10.89%	42.57%	100%
Text 3 (C1 level)	TEXT TOKENS: 499 HUMAN ANNOTATIONS: 69 GPT'S ANNOTATIONS: 51	12	30	7	32	81
		14.81%	37.04%	8.64%	39.51%	100%
All texts	TEXT TOKENS 1,250 HUMAN ANNOTATIONS: 230 GPT'S ANNOTATIONS: 144	21	107	27	96	251
		8.37%	**42.63%**	**10.75%**	**38.25%**	**100%**

Table 19 – Accuracy results for full-text annotation

As for the overall percentage of missed tags (42.63%), its value remains high likely because of the complexity of the task, especially when subtle errors are to be detected, since this requires an extensive effort of sentence interpretation, for instance:

HUMAN ANNOTATOR: <DMCC corr="I">*i*</DMCC> want to <**LSV** corr="ask">*put*</**LSV**> you some **<LSN corr="questions">***answers***</LSN>**

GPT: <DMCC corr="I">*i*</DMCC> want to <**FS** corr="ask">*put*</**FS**> you some answers about it

In this example, the GPT fails to correctly annotate the lexical issues: while the sentence is grammatically correct, the verb 'put' is misclassified under Form Spelling (<FS>) instead of Lexis Single Verb (<LSV>), and the use of the incorrect noun 'answers' is not detected at all, resulting in a nonsensical sentence. Finally, looking at Table 19, an interesting pattern emerges: with higher proficiency levels (B2 and C2), the percentage of partially correct annotations decreases, while the percentage of correct annotations increases.

6. Discussion and Conclusions

6.1. Summary of Results

This thesis was conducted within the *UNITE — Universally inclusive technologies to practice English* project[21], which aims to create and analyse a learner corpus based on student-chatbot interactions, with the ultimate goal of supporting the effective integration of AI technology in language education. Specifically, this research focused on streamlining the error annotation step of the UNITE project through two case studies: the first aimed to identify the most suitable error taxonomy for the conversational nature of the UNITE corpus, while the second investigated the possibility of automating the error annotation process using ChatGPT.

The first case study involved annotating errors in students' conversational turns from a sample of 23 texts within the UNITE corpus. The error taxonomy used was the one described in the Louvain Error Tagging Manual Version 2.0 (Granger *et al.*, 2022), which classifies errors into seven macro-categories: Form, Grammar, Lexico-Grammar, Lexis, Word, Punctuation, and Infelicities. The annotation task was conducted in three rounds, with each round refining the methodology based on insights from the previous rounds. In the first round, the Louvain manual was applied verbatim, except for two modifications: the addition of the Code-Switching category and the exclusion of certain elements from the Infelicities category (e.g., slang, abbreviations, and informal expressions). From this round, it emerged that features typically associated with Digitally-Mediated-Communication (DMC) (e.g., uncapitalized first-person singular pronoun) were influencing data on error distribution, as they were originally categorised as spelling errors under the Form category. To address this imbalance, the second-round annotation excluded DMC features from the annotated dataset. The third-round annotation implemented the final version of the error taxonomy, incorporating: (1) a new macro-category for DMC features, (2) two subcategories for Code-Switching to differentiate between intra-sentential and inter-sentential occurrences, and (3) an additional subcategory for lexical coinages and calques within the Lexis macro-category.

[21] https://site.unibo.it/unite/en

The error distribution analysis was based on the third-round annotation, with error frequencies normalised per 1,000 tokens (ptt). The analysis was conducted at three different levels: (1) across the entire sample corpus, (2) by English proficiency level, and (3) comparatively across English proficiency levels. Overall, within the entire annotated dataset, results showed that the most frequent error category was DMC, followed by Grammar and Form. Moreover, even if not among the most prevalent error types, the Code-Switching category proved to be slightly more frequent than Lexico-Grammar and Infelicities. At a broader level of categorisation, orthographic and morphological errors (Form and Punctuation) emerged as the most prominent linguistic area, followed closely by pragmatic and stylistic errors (DMC, Code-Switching, and Infelicities).

In respect to error distribution by English level, the observed trend confirmed the expected decrease in errors as proficiency increases, with two exceptions: a slight peak at A2 level compared to A1, and a considerable increase at C2, where error frequency was higher than at both B2 and C1.

Regarding error distribution across English levels, the trends for each error category showed that all categories aligned to some extent with the global trend, with peaks at A2 and/or C2 level. However, the most notable trends were observed in the DMC and Form categories. In both cases, the peak at the C2 level was substantially higher compared to the other error categories.

Moving to the second case study, it served as a proof-of-concept experiment involving three main steps: (1) converting the Louvain manual into Markdown format for chatbot compatibility and adapting its content to reflect the refined error taxonomy, (2) creating a custom GPT for error annotation, and (3) testing the custom GPT's performance. The first step was essential to create a "chatbot-friendly" version of the manual and make sure that the model could effectively reference it during the annotation task. The second step involved testing different ChatGPT conversational modes: basic Chat, Project, and custom GPT. The latter became the final choice as it emerged as the most accurate due to the possibility of uploading the manual as a knowledge base and fine-tuning the model through detailed inner instructions. The third step consisted in testing the custom GPT using two approaches: step-by-step annotation (three to five conversational turns at a time) and full-text annotation (entire text in one go). While only one text was annotated using the step-by-step method due to its time-consuming nature, three texts were annotated using the full-text approach.

The custom GPT's output was then analysed to assess its level of accuracy, using human annotations as the gold standard and grouping the GPT's annotations in four categories: correct, partially correct, missed, and incorrect. The partially correct category included annotations where at least two elements—correction, incorrect text, or tag—matched the gold standard (e.g., CORRECTION + INCORRECT TEXT), as well as cases where only the incorrect text matched, while the tag and correction differed.

The results showed that, in both annotation approaches, the GPT produced fewer annotations than the human annotator, scoring an accuracy range between 30% and 42%. Unexpectedly, the full-text approach outperformed the step-by-step approach (38.25% global accuracy vs. 33.33%, respectively). However, it is important to remember that, since the two methods were tested on different sample sizes, further testing would be needed. Moreover, the analysis of partially correct instances revealed that, out of 27 instances, only three were cases of same TAG + CORRECTION, meaning that in all other cases the GPT adhered to tag placement rules but applied a different tag or correction. Finally, another important observation emerged when analysing the accuracy of each text annotated using the full-text approach: the percentage of correct instances increased at higher proficiency levels—in this case B2 and C2 texts achieving an accuracy of 42.47% and 39.51%, respectively—compared to the A1 text (30.43%).

6.2. Discussion

With respect to the results on error distribution, the fact of DMC emerged as the most frequent category, followed by Grammar and Form category, highlights the importance of having a separate category for DMC features. Indeed, as aforementioned, having them categorised under Form was heavily influencing error distribution, making Form the most frequent category in the first-round annotation.

As for the prevalence of Code-Switching instances over Lexico-Grammar and Infelicities, it might suggest that, in this specific conversational context, when learners experience difficulty in communicating, they may feel more at ease in switching to their mother tongue instead of attempting a possibly incorrect English structure. This tendency may be influenced by the learners' awareness that the chatbot can understand them also in their native language. This insight, alongside with the previous one, confirm the necessity of using a refined the error taxonomy to ensure an accurate categorisation of the linguistic characteristics of the UNITE corpus.

In terms of error distribution by English level, the observed considerable increase at C2 may be attributed to learners' greater linguistic confidence, which leads them to: (1) attempt more complex linguistic structures (MacDonald, 2016: 123), (2) adopt a more relaxed behaviour when interacting with the chatbot. Indeed, this may result in a higher number of errors in specific categories such as DMC and Word. Nevertheless, it is also worth remembering that the self-assessed nature of the learners' proficiency level is an important factor to be considered when interpreting these results, since the learners' perception on their proficiency might not always align with their actual level (see Section 6.3), which may also explain the slight increase at A2 level.

Regarding the most notable trends observed in error frequency across English levels, the substantially higher peak at the C2 for both DMC and Form categories aligns with the earlier hypothesis that advanced learners' greater linguistic confidence may result in a more relaxed behaviour during the interaction. Indeed, this attitude may lead in more typos and intentional use of DMC features. Similarly, the substantial peak at the C2 level for the Word category, highlights once again the possible attempt of advanced learners to use more complex wording that may contribute to a higher frequency of errors.

Moving to the results of the second case study, it is pivotal not only to look at the GPT's accuracy value, but also to the number and nature of partially incorrect instances. Indeed, as mentioned in the previous section, the results indicate that the GPT generally follows tag placement rules, suggesting that it shows more difficulties in assigning the correct tag rather than spotting the error. This difficulty is likely due to the large number of tags in the tagset (59 in total). However, considered that tag placement was one of the main issues encountered when testing other ChatGPT conversational modes, these findings highlight that using the custom GPT with the refined manual in Markdown as a knowledge base proved to be particularly effective in ensuring accurate tag placement.

The other significant finding emerged from this analysis is that the GPT's percentage of correct annotations increased at higher proficiency levels. This could be explained by the reduced number and/or gravity of errors in texts produced by higher proficiency students, with the chatbot being able to spot a higher percentage of errors. Actually, it is not to be excluded that texts with a higher density of linguistic errors may confuse the chatbot during the annotation process. It goes without saying that this hypothesis should be confirmed by further testing on a larger sample of texts. Nevertheless, knowing that this could be a possible pattern, this insight would help human annotators focus more on reviewing chatbot's annotations of lower proficiency texts.

Finally, it is worth remembering that the chatbot may occasionally hallucinate, such as by adding full stops or capital letters even when not present in the original text or by detecting non-errors (false positives) and, as aforementioned, misclassifying errors by confusing one tag with another. This implies extra attention on the part of human annotators when using the chatbot as a preliminary tool for error annotation, both in ensuring that the original text remains unaltered and in making sure that the correct error tag is applied. A similar issue is noted in Imamovic *et al.* (2024: 119), meaning that this represents a common problem in the use of ChatGPT for annotation tasks. For this reason, if using this tool, accurate revision and post-editing of the chatbot's annotations are strongly recommended.

6.3. Limitations and Future Work

The findings of the two case studies should be considered in the light of some limitations. First, both the manual annotation task and the testing of the custom GPT were conducted on a small sample of texts. Therefore, even though this research does give a first insight into error distribution and the possibility of automating the error annotation process to some extent, a larger sample of texts would be necessary to confirm or disprove these results.

Second, as mentioned in the previous section, the self-assessed nature of the learners' level of proficiency in English may influence the data on error distribution, since their perceived level may not always align with their actual proficiency. Indeed, research showed that it is not infrequent that students may feel overconfident and overestimate their abilities (Petersen, 2018).

Third, as for the use and testing of the custom GPT for annotation, it should be borne in mind that the gold standard used for calculating the GPT's accuracy is based on the annotations of a single human annotator. Since error annotation is inherently subject to personal interpretation (Granger, 2003: 475), involving multiple annotators would help to establish a more reliable gold standard. This would provide a more robust basis for the evaluation of the chatbot's output.

Moreover, a possible solution for improving the chatbot's accuracy in future work would be to use a simplified error taxonomy (e.g., by relying on broader subcategories). Indeed, this simplification might help the chatbot choose and apply the correct tag. Alternatively, accuracy could be improved by refining the adapted error annotation manual, for example through the insertion of additional explanatory tables for "fuzzy" areas (this approach already proved useful as described in Section 5.2.1). Indeed, with further testing, it may be possible to identify which tags the chatbot tends to confuse more often and to implement targeted adjustments.

To conclude, while this case study demonstrated ChatGPT's ability to handle the complex task of error annotation by identifying a substantial number of errors, future research should focus on improving its accuracy. This could be done by testing the presented method on a larger dataset, applying the suggested refinements, or using alternative approaches, such as testing other chatbots or training an LLM on an annotated dataset.

References

Agustini, N. P. O. (2023). ‘Examining the Role of ChatGPT as a Learning Tool in Promoting Students’ English Language Learning Autonomy Relevant to Kurikulum Merdeka Belajar’, *EDUKASIA: Jurnal Pendidikan Dan Pembelajaran* 4(2): 921–934. https://doi.org/10.62775/edukasia.v4i2.373.

Alemi, M., Meghdari, A., and Ghazisaedy, M. (2015). ‘The Impact of Social Robotics on L2 Learners’ Anxiety and Attitude in English Vocabulary Acquisition’, *International Journal of Social Robotics* 7(4): 523–535. https://doi.org/10.1007/s12369-015-0286-y.

Aliyar, M. (2020). ‘A Corpus-Based Analysis of Linguistic Interference of Italian Native Speakers in English and Spanish Written Production as a Foreign Language’, *Journal of Second Language Teaching & Research* 8(1): 103–137. https://pops.uclan.ac.uk/index.php/jsltr/article/view/592 [Accessed: 25 February 2025].

Anderson, B. (2024). ‘ChatGPT vs Pi – Personal AI’. *SEO Rocket Blog*. https://blog.seorocket.ai/chatgpt-vs-pi-personal-ai/ [Accessed: 20 January 2025].

Anderson, J. N., Davidson, N., Morton, H., and Jack, M. A. (2008). ‘Language Learning with Interactive Virtual Agent Scenarios and Speech Recognition: Lessons Learned’, *Computer Animation and Virtual Worlds* 19(5): 605–619. https://doi.org/10.1002/cav.265.

Anthony, L. (2024). *A Comprehensive Guide to AntConc 4: New Tools, Features, and AI Integration*. https://osf.io/euryh [Accessed: 16 February 2025].

Athaluri, S. A. *et al.* (2023). ‘Exploring the Boundaries of Reality: Investigating the Phenomenon of Artificial Intelligence Hallucination in Scientific Writing Through ChatGPT References’, *Cureus* 15(4): e37432. https://doi.org/10.7759/cureus.37432.

Belda-Medina, J., and Calvo-Ferrer, J. R. (2022). ‘Using Chatbots as AI Conversational Partners in Language Learning’, *Applied Sciences* 12(17): 8427. https://doi.org/10.3390/app12178427.

Bibauw, S., François, T., and Desmet, P. (2015). ‘Dialogue-Based CALL: An Overview of Existing Research’, in F. Helm, L. Bradley, M. Guarda, and S. Thouësny (eds). *Critical CALL – Proceedings of the 2015 EUROCALL Conference Padova, Italy*, 2 December 2015, Dublin: Research-publishing.net, 57–64. https://doi.org/10.14705/rpnet.2015.000310.

Bibauw, S., François, T., and Desmet, P. (2019). ‘Discussing with a Computer to Practice a Foreign Language: Research Synthesis and Conceptual Framework of Dialogue-Based CALL’, *Computer Assisted Language Learning, [FirstView]*. https://doi.org/10.1080/09588221.2018.1535508.

Bibauw, S., François, T., and Desmet, P. (2022a). ‘Dialogue Systems for Language Learning: Chatbots and Beyond’, in *The Routledge Handbook of Second Language Acquisition and Technology*, New York: Routledge, 121–135. https://doi.org/10.4324/9781351117586-12.

Bibauw, S., Van den Noortgate, W., François, T., and Desmet, P. (2022b). ‘Dialogue Systems for Language Learning: A Meta-Analysis’, *Language Learning & Technology* 26(1): 1–24. https://www.lltjournal.org/item/10125-73488/.

Caldarini, G., Jaf, S., and McGarry, K. (2022). ‘A Literature Survey of Recent Advances in Chatbots’, *Information* 13(41): 1–22. https://doi.org/10.3390/info13010041.

Cambridge University Press (2006). *Cambridge Learner Corpus*.

Centre for English Corpus Linguistics (2023). *Learner Corpora around the World*, Louvain-la-Neuve: Université catholique de Louvain.

Cervini, C., and Paone, E. (2024). ‘COMUNICARE ALL’UNIVERSITÀ: QUANDO L’INTERAZIONE ORALE SI FA PLURILINGUE’, *Italiano LinguaDue* 16(2): 496–523. https://doi.org/10.54103/2037-3597/27861.

Cone, M. (2024). *Getting Started - Markdown Guide*. https://www.markdownguide.org/getting-started/ [Accessed: 20 January 2025].

Copeland, B. J. (2025). ‘Artificial Intelligence (AI)’. *Britannica*. https://www.britannica.com/technology/artificial-intelligence [Accessed: 10 February 2025].

Dagneaux, E., Denness, S., and Granger, S. (1998). ‘Computer-Aided Error Analysis’, *System: An International Journal of Educational Technology and Applied Linguistics* 26(2): 163–174. https://doi.org/10.1016/S0346-251X(98)00001-3.

Dang, H., Mecke, L., Lehmann, F., Goller, S., and Buschek, D. (2022). *How to Prompt? Opportunities and Challenges of Zero- and Few-Shot Learning for Human-AI Interaction in Creative Applications of Generative Models*. https://doi.org/10.48550/arXiv.2209.01390.

Davis, C. *et al.* (2024). *Prompting open-source and commercial language models for grammatical error correction of English learner text*. https://doi.org/10.48550/ARXIV.2401.07702.

DeVon, C. (2023). ‘On ChatGPT’s One-Year Anniversary, It Has More than 1.7 Billion Users—Here’s What It May Do Next’, *CNBC* 30 November. https://www.cnbc.com/2023/11/30/chatgpts-one-year-anniversary-how-the-viral-ai-chatbot-has-changed.html [Accessed: 23 February 2025].

Díaz-Negrillo, A., and Fernández-Domínguez, J. (2006). ‘Error Tagging Systems for Learner Corpora’, *Revista Española de Lingüística Aplicada* 19: 83–102. https://www.researchgate.net/publication/28137922_Error_Tagging_Systems_for_Learner_Corpora.

Díaz-Negrillo, A., and García-Cumbreras, M. Á. (2007). ‘A Tagging Tool for Error Analysis on Learner Corpora’, *ICAME* 31: 197–203. http://clu.uni.no/icame/ij31/ij31-page197-204.pdf [Accessed: 14 February 2025].

Dokukina, I., and Gumanova, J. (2020). ‘The Rise of Chatbots – New Personal Assistants in Foreign Language Learning’, *Procedia Computer Science* 169: 542–546. https://doi.org/10.1016/j.procs.2020.02.212.

Dulay, H. C., Burt, M., and Krashen, S. (1982). *Language Two*, New York: Oxford University Press.

Fryer, L. K., Ainley, M., Thompson, A., Gibson, A., and Sherlock, Z. (2017). ‘Stimulating and Sustaining Interest in a Language Course: An Experimental Comparison of Chatbot and Human Task Partners’, *Computers in Human Behavior* 75: 461–468. https://doi.org/10.1016/j.chb.2017.05.045.

Fryer, L. K., and Carpenter, R. (2006). ‘Bots as Language Learning Tools’, *Language Learning & Technology* 10(3): 8–14. https://www.lltjournal.org/item/10125-44068/ [Accessed: 16 February 2025].

Gillis, A. S. (2024). *What Is Robotic Process Automation (RPA)?* TechTarget. https://www.techtarget.com/searchcio/definition/RPA [Accessed: 16 February 2025].

Gilquin, G. (2010). *The adventure of international learner corpora: Implications and applications*,.

Gilquin, G. (2015). ‘From Design to Collection of Learner Corpora’, in F. Meunier, G. Gilquin, and S. Granger (eds). *The Cambridge Handbook of Learner Corpus Research*, Cambridge: Cambridge University Press, 9–34. https://doi.org/10.1017/CBO9781139649414.002.

Gilquin, G. (2020). ‘Learner Corpora’, in M. Paquot and S. Th. Gries (eds). *A Practical Handbook of Corpus Linguistics*, Cham: Springer, 283–303.

Gilquin, G., De Cock, S., and Granger, S. (2010). *The Louvain International Database of Spoken English Interlanguage. Handbook and CD-ROM*, Louvain-la-Neuve: Presses universitaires de Louvain.

Gobiet, M. (2024). *The History Of Chatbots – From ELIZA to ChatGPT*. 15 February. Onlim. https://onlim.com/en/the-history-of-chatbots/ [Accessed: 9 February 2025].

Google (2024). *Generate Text, Images, Code, and More with Google Cloud AI*. Google Cloud. https://cloud.google.com/use-cases/generative-ai [Accessed: 9 February 2025].

Granger, S. (2003). 'Error-Tagged Learner Corpora and CALL: A Promising Synergy', *CALICO* 20(3): 465–480. https://www.jstor.org/stable/24157525.

Granger, S. (2015). 'Contrastive Interlanguage Analysis: A Reappraisal', *International Journal of Learner Corpus Research* 1: 7–24. https://doi.org/10.1075/ijlcr.1.1.01gra.

Granger, S. (2017). 'Learner Corpora in Foreign Language Education', in S. L. Thorne and S. May (eds). *Language, Education and Technology*, Cham: Springer, 427–440. https://doi.org/10.1007/978-3-319-02237-6_33.

Granger, S., Dagneaux, E., and Meunier, F. (2002). *The International Corpus of Learner English: Handbook and CD-ROM*, Louvain-la-Neuve: Presses universitaires de Louvain.

Granger, S., Dagneaux, E., Meunier, F., and Paquot, M. (2009). *The International Corpus of Learner English. Handbook and CD-ROM. Version 2*, Louvain-la-Neuve: Presses universitaires de Louvain.

Granger, S., Maïté, D., Fanny, M., Hubert, N., and Magali (2020). *The International Corpus of Learner English. Version 3*, Louvain-la-Neuve: Presses universitaires de Louvain.

Granger, S., Swallow, H., and Thewissen, J. (2022). *The Louvain Error Tagging Manual Version 2.0*. https://oer.uclouvain.be/jspui/bitstream/20.500.12279/968/4/Granger%20et%20al._Error%20tagging%20manual%202.0_final_CC.pdf [Accessed: 17 September 2024].

Granger, S., Swallow, H., and Thewissen, J. (2023). *The UCLouvain Error Editor User guide - Version 2.0*. https://oer.uclouvain.be/jspui/bitstream/20.500.12279/968/5/UCLEE%20user%20guide_final-CC.pdf.

Houghton, K. J., Upadhyay, S. S. N., and Klin, C. M. (2018). 'Punctuation in Text Messages May Convey Abruptness. Period', *Computers in Human Behavior* 80: 112–121. https://doi.org/10.1016/j.chb.2017.10.044.

Huang, W., Hew, K. F., and Fryer, L. K. (2022). ‘Chatbots for Language Learning—Are They Really Useful? A Systematic Review of Chatbot-supported Language Learning’, *Journal of Computer Assisted Learning* 38(1): 237–257. https://doi.org/10.1111/jcal.12610.

Hubbard, P. (2021). *An Invitation to CALL: Foundations of Computer-Assisted Language Learning*, APACALL.

IBM (2021a). *What Is a Chatbot?*. https://www.ibm.com/think/topics/chatbots [Accessed: 8 February 2025].

IBM (2021b). *What Is Machine Learning (ML)?*. https://www.ibm.com/think/topics/machine-learning [Accessed: 9 February 2025].

IBM (2021c). *What Is Robotic Process Automation (RPA)?*. https://www.ibm.com/think/topics/rpa [Accessed: 10 February 2025].

IBM (2023). *What Are Large Language Models (LLMs)?*. https://www.ibm.com/think/topics/large-language-models [Accessed: 9 February 2025].

IBM (2024). *Types of Chatbots*. https://www.ibm.com/think/topics/chatbot-types [Accessed: 16 February 2025].

Imamovic, M., Deilen, S., Glynn, D., and Lapshinova-Koltunski, E. (2024). ‘Using ChatGPT for Annotation of Attitude within the Appraisal Theory: Lessons Learned’, in S. Henning and M. Stede (eds). *Proceedings of The 18th Linguistic Annotation Workshop (LAW-XVIII)*, March 2024, St. Julians, Malta: Association for Computational Linguistics, 112–123. https://aclanthology.org/2024.law-1.11/ [Accessed: 20 January 2025].

Izumi, E., Saiga, T., Supnithi, T., Uchimoto, K., and Isahara, H. (2003). ‘The Development of the Spoken Corpus of Japanese Learner English and the Applications in Collaboration with NLP Techniques’, in D. Archer, P. Rayson, W. Andrew, and T. McEnery (eds). *Proceedings of the Corpus Linguistics 2003 Conference*, United Kingdom: Lancaster University, 359–566.

Izumi, E., Uchimoto, K., and Isahara, H. (2005). ‘Error Annotation for Corpus of Japanese Learner English’, in *Proceedings of the Sixth International Workshop on Linguistically Interpreted Corpora (LINC 2005)*, 15 October 2005, Jesu Island, Korea, 1–80. http://acl.ldc.upenn.edu/I/I05/I05-6009.pdf.

Jia, J., Chen, Y., Ding, Z., and Ruan, M. (2012). 'Effects of a Vocabulary Acquisition and Assessment System on Students' Performance in a Blended Learning Class for English Subject', *Computers & Education* 58(1): 63–76. https://doi.org/10.1016/j.compedu.2011.08.002.

Kaplan, J. D., Sabol, M. A., Wisher, R. A., and Seidel, R. J. (1998). 'The Military Language Tutor (MILT) Program: An Advanced Authoring System', *Computer Assisted Language Learning* 11(3): 265–287. https://doi.org/10.1076/call.11.3.265.5679.

Kumar, E. S., and Sreehari, P. (2009). 'Computer Assisted Language Learning (CALL)', in *A Handbook for English Language Laboratories*, Foundation Books, 1–2. https://doi.org/10.1017/UPO9788175968677.

Law, L. (2024). 'Application of Generative Artificial Intelligence (GenAI) in Language Teaching and Learning: A Scoping Literature Review', *Computers and Education Open* 6: 100174. https://doi.org/10.1016/j.caeo.2024.100174.

Lester, J. C. *et al.* (1997). 'The Persona Effect: Affective Impact of Animated Pedagogical Agents', in *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, 27 March 1997, New York, NY, USA: Association for Computing Machinery, 359–366. https://doi.org/10.1145/258549.258797.

L'haire, S., and Vandeventer-Faltin, A. (2003). 'Diagnostic d'erreurs dans le projet FreeText', *Alsic. Apprentissage des Langues et Systèmes d'Information et de Communication* 6(2): 21–37. https://doi.org/10.4000/alsic.2219.

Lüdeling, A., and Hirschmann, H. (2015). 'Error Annotation Systems', in S. Granger, G. Gilquin, and F. Meunier (eds). *The Cambridge Handbook of Learner Corpus Research*, Cambridge: Cambridge University Press, 135–158. https://doi.org/10.1017/CBO9781139649414.007.

Luo, B., Lau, R. Y. K., Li, C., and Si, Y.-W. (2022). 'A Critical Review of State-of-the-Art Chatbot Designs and Applications', *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 12(1). https://doi.org/10.1002/widm.1434.

Lyddy, F., Farina, F., Hanney, J., Farrell, L., and Kelly O'Neill, N. (2014). 'An Analysis of Language in University Students' Text Messages', *Journal of Computer-Mediated Communication* 19(3): 546–561. https://doi.org/10.1111/jcc4.12045.

MacDonald, P. (2016). '"We All Make Mistakes!". Analysing an Error-Coded Corpus of Spanish University Students' Written English', *Complutense Journal of English Studies* 24(26): 103–129. https://doi.org/10.5209/CJES.53273.

Mageira, K. *et al.* (2022). 'Educational AI Chatbots for Content and Language Integrated Learning', *Applied Sciences* 12(7): 3239. https://doi.org/10.3390/app12073239.

McEnery, T., Love, R., and Brezina, V. (2017). 'Compiling and Analysing the Spoken British National Corpus 2014', *International Journal of Corpus Linguistics* 22(3): 311–318. https://doi.org/10.1075/ijcl.22.3.01mce.

Meunier, F. (2020). 'Introduction to Learner Corpus Research', in N. Tracy-Ventura and M. Paquot (eds). *The Routledge Handbook of Second Language Acquisition and Corpora*, New York: Routledge, 23–36.

Meunier, F., Melo, M., and Dalipi, F. (2021). *The Dawn of the Human-Machine Era: A Forecast of New and Emerging Language Technologies*.

Mirani, J. I., Lohar, S. A., Jat, A. R. L., and Faheem, M. (2019). 'A Review of Computer-Assisted Language Learning (CALL): Development, Challenges, and Future Impact', *Education and Linguistics Research* 5(1): 37. https://doi.org/10.5296/elr.v5i1.14515.

Nesselhauf, N. (2004). 'Learner Corpora and Their Potential for Language Teaching', in J. McH. Sinclair (ed.). *How to Use Corpora in Language Teaching*, John Benjamins, 125–152. https://doi.org/10.1075/scl.12.11nes.

Nicolaidou, I., Pissas, P., and Boglou, D. (2023). 'Comparing Immersive Virtual Reality to Mobile Applications in Foreign Language Learning in Higher Education: A Quasi-Experiment', *Interactive Learning Environments* 31(4): 2001–2015. https://doi.org/10.1080/10494820.2020.1870504.

OpenAI (2024a). *Introducing GPT-4o and More Tools to ChatGPT Free Users*. https://openai.com/index/gpt-4o-and-more-tools-to-chatgpt-free/ [Accessed: 20 January 2025].

OpenAI (2024b). *Key Guidelines for Writing Instructions for Custom GPTs*. https://help.openai.com/en/articles/9358033-key-guidelines-for-writing-instructions-for-custom-gpts [Accessed: 20 January 2025].

OpenAI (2024c). *Using Projects in ChatGPT*. https://help.openai.com/en/articles/10169521-using-projects-in-chatgpt [Accessed: 20 January 2025].

OpenAI (2025). *GPTs FAQ*. https://help.openai.com/en/articles/8554407-gpts-faq [Accessed: 20 January 2025].

Oppy, G., and Dowe, D. (2021). 'The Turing Test' E. N. Zalta (ed.). *The Stanford Encyclopedia of Philosophy*. https://plato.stanford.edu/archives/win2021/entriesuring-test/ [Accessed: 9 February 2025].

Pérez-Marín, D. (2021). 'A Review of the Practical Applications of Pedagogic Conversational Agents to Be Used in School and University Classrooms', *Digital* 1(1): 18–33. https://doi.org/10.3390/digital1010002.

Petersen, J. M. (2018). *Performance Self-Appraisal Calibration of ESL Students on a Proficiency Reading Test*. Brigham Young University. https://scholarsarchive.byu.edu/etd/6764/?utm_source=scholarsarchive.byu.edu%2Fetd%2F6764&utm_medium=PDF&utm_campaign=PDFCoverPages [Accessed: 2 February 2025].

Poplack, S. (1980). 'Sometimes I'll Start a Sentence in Spanish Y TERMINO EN ESPAÑOL: Toward a Typology of Code-Switching', *Linguistics* 18(7/8): 581–618. https://doi.org/10.1515/ling.1980.18.7-8.581.

Ramanathan, T. (2025). 'Natural Language Processing (NLP)'. *Britannica*. https://www.britannica.com/technology/natural-language-processing-computer-science [Accessed: 10 February 2025].

Rethmeier, N. (2011). *Using Language Models to Detect Errors in Second-Language Learner Writing*. Bauhaus-Universität Weimar. https://downloads.webis.de/theses/papers/rethmeier_2011.pdf [Accessed: 16 February 2025].

Selinker, L. (1972). 'INTERLANGUAGE', *International Review of Applied Linguistics in Language Teaching* 10(1–4): 209–232. https://doi.org/10.1515/iral.1972.10.1-4.209.

Smutny, P., and Schreiberova, P. (2020). 'Chatbots for Learning: A Review of Educational Chatbots for the Facebook Messenger', *Computers & Education* 151: 103862. https://doi.org/10.1016/j.compedu.2020.103862.

Solis-Quispe, J. M., Quico-Cauti, K. M., and Ugarte, W. (2021). 'Chatbot to Simplify Customer Interaction in E-Commerce Channels of Retail Companies', in *Information Technology and Systems*, 2021, Cham: Springer, 561–570. https://doi.org/10.1007/978-3-030-68285-9_52.

Tagg, C. (2012). *Discourse of Text Messaging: Analysis of SMS Communication*, London: Continuum.

Turing, A. (1950). 'Computing Machinery and Intelligence', *Mind* LIX(236): 433–460. https://doi.org/10.1093/mind/LIX.236.433.

UNITE (2024). https://site.unibo.it/unite/en [Accessed: 17 September 2024].

Valdivieso Castillo, P. Á., and Aguilar Luzon, M. del C. (2021). 'The Use of Chatbot as an Element of Tutorial Action in University Teaching', *REIDOCREA* 10(24): 1–14. https://doi.org/10.30827/Digibug.69299.

Verheijen, L. (2015). 'Out-of-the-Ordinary Orthography: The Use of Textisms in Dutch Youngsters' Written Computer-Mediated Communication', in *PARLAY Proceedings*, January 2015, York, England, 127–142. https://hdl.handle.net/2066/151594 [Accessed: 20 January 2025].

Wang, F., Cheung, A. C. K., Neitzel, A. J., and Chai, C. S. (2024). 'Does Chatting with Chatbots Improve Language Learning Performance? A Meta-Analysis of Chatbot-Assisted Language Learning', *Review of Educational Research* 20(10): 1–38. https://doi.org/10.3102/00346543241255621.

Wang, N., and Johnson, W. L. (2008). 'The Politeness Effect in an Intelligent Foreign Language Tutoring System', in B. P. Woolf, E. Aïmeur, R. Nkambou, and S. Lajoie (eds). *Intelligent Tutoring Systems*, 2008, Berlin, Heidelberg: Springer, 270–280. https://doi.org/10.1007/978-3-540-69132-7_31.

Wood, D., Achhpilia, M. P., Adams, M. T., and Aghazadeh, S. (2023). 'The ChatGPT Artificial Intelligence Chatbot: How Well Does It Answer Accounting Assessment Questions?', *Issues in Accounting Education* 38(4): 1–28. https://doi.org/10.2308/ISSUES-2023-013.

Xu, Y. *et al.* (2022). 'Dialogue with a Conversational Agent Promotes Children's Story Comprehension via Enhancing Engagement', *Child Development* 93(2): e149–e167. https://doi.org/10.1111/cdev.13708.

Xu, Y., Wang, D., Collins, P., Lee, H., and Warschauer, M. (2021). 'Same Benefits, Different Communication Patterns: Comparing Children's Reading with a Conversational Agent vs. a Human Partner', *Computers & Education* 161: 104059. https://doi.org/10.1016/j.compedu.2020.104059.

Yan, D. (2023). 'Impact of ChatGPT on Learners in a L2 Writing Practicum: An Exploratory Investigation', *Education and Information Technologies* 28(11): 13943–13967. https://doi.org/10.1007/s10639-023-11742-4.

Yang, L., and Li, R. (2024). 'ChatGPT for L2 Learning: Current Status and Implications', *System* 124: 103351. https://doi.org/10.1016/j.system.2024.103351.

Yin, Q., and Satar, M. (2020). 'English as a Foreign Language Learner Interaction with Chatbots: Negotiation for Meaning', *International Online Journal of Education and Teaching (IOJET)* 7(2): 390–410. http://iojet.org/index.php/IOJET/article/view/707.

Yu, D., Li, L., Su, H., and Fuoli, M. (2024). 'Assessing the Potential of LLM-Assisted Annotation for Corpus-Based Pragmatics and Discourse Analysis: The Case of Apology', *International Journal of Corpus Linguistics* 29(4): 534–561. https://doi.org/10.1075/ijcl.23087.yu.

Yudytska, J. (2024). 'The Interaction of Technological Affordances and User Preferences: A Corpus-Based Study of Graphic Features across Twitter and Discord', *Discourse, Context & Media* 62: 100836. https://doi.org/10.1016/j.dcm.2024.100836.

Zemčík, T. (2019). 'A Brief History of Chatbots', in *DEStech Transactions on Computer Science and Engineering*, 23 June 2019, Wuhan, China, 14–18. https://doi.org/10.12783/dtcse/aicae2019/31439.

Zhai, C., and Wibowo, S. (2023). 'A Systematic Review on Artificial Intelligence Dialogue Systems for Enhancing English as Foreign Language Students' Interactional Competence in the University', *Computers and Education: Artificial Intelligence* 4: 100134. https://doi.org/10.1016/j.caeai.2023.100134.

Appendix

A. Post-interaction Questionnaire (Italian with English Translation)

Metadati *Metadata*

1) Indica il tuo indirizzo e-mail *Insert your email address*

2) A quale corso di laurea sei iscritto/a? OPPURE: Il corso di laurea a cui sei iscritt* appartiene all'area... *Which degree programme are you enrolled in? OR: The degree programme in which you are enrolled belongs to the field of...*

- Umanistica *Humanities*
- Politico-sociale *Political and Social sciences*
- Giuridica *Law*
- Economica *Economics*
- Scientifica *Science*
- Tecnologica *Technology*
- Sanitaria *Healthcare*

3) Età *Age*

4) Genere *Gender*

- Uomo *Male*
- Donna *Female*
- Non binario *Non-binary*
- Preferisco non dire *Prefer not to say*
- Altro *Other*

5) Qual è la tua lingua madre? *What is your native language?*

6) Hai disabilità e/o DSA (disturbi specifici dell'apprendimento)? *Do you have any disability and/or SLD (specific learning disorder)?*

6.1) Quali? *Which ones?*

7) Come valuteresti il tuo livello per quanto riguarda la comprensione scritta di testi in inglese? I livelli fanno riferimento al Quadro comune europeo di riferimento per le lingue (CEFR). *How would you assess your level of reading comprehension in English? The levels refer to the Common European Framework of Reference for Languages (CEFR).*

- Livello base (A1) *Beginner level (A1)*
- Livello elementare (A2) *Elementary level (A2)*
- Livello intermedio (B1) *Intermediate level (B2)*
- Livello intermedio superiore (B2) *Upper intermediate level (B1)*
- Livello avanzato (C1) *Advanced level (C1)*
- Livello di padronanza (C2) *Proficient level (C2)*

8) Come valuteresti il tuo livello per quanto riguarda la produzione scritta di testi in inglese? I livelli fanno riferimento al Quadro comune europeo di riferimento per le lingue (CEFR) *How would you assess your level of written production in English? The levels refer to the Common European Framework of Reference for Languages (CEFR).*

- Livello base (A1) *Beginner level (A1)*
- Livello elementare (A2) *Elementary level (A2)*
- Livello intermedio (B1) *Intermediate level (B2)*
- Livello intermedio superiore (B2) *Upper intermediate level (B1)*
- Livello avanzato (C1) *Advanced level (C1)*
- Livello di padronanza (C2) *Proficient level (C2)*

9) Come valuteresti il tuo livello per quanto riguarda la comprensione orale di testi in inglese? I livelli fanno riferimento al Quadro comune europeo di riferimento per le lingue (CEFR) *How would you assess your level of oral comprehension in English? The levels refer to the Common European Framework of Reference for Languages (CEFR).*

- Livello base (A1) *Beginner level (A1)*
- Livello elementare (A2) *Elementary level (A2)*
- Livello intermedio (B1) *Intermediate level (B2)*
- Livello intermedio superiore (B2) *Upper intermediate level (B1)*
- Livello avanzato (C1) *Advanced level (C1)*
- Livello di padronanza (C2) *Proficient level (C2)*

10) Come valuteresti il tuo livello per quanto riguarda la produzione orale di testi in inglese? I livelli fanno riferimento al Quadro comune europeo di riferimento per le lingue (CEFR) *How would you assess your level of oral production in English? The levels refer to the Common European Framework of Reference for Languages (CEFR).*

- Livello base (A1) *Beginner level (A1)*
- Livello elementare (A2) *Elementary level (A2)*
- Livello intermedio (B1) *Intermediate level (B2)*
- Livello intermedio superiore (B2) *Upper intermediate level (B1)*
- Livello avanzato (C1) *Advanced level (C1)*
- Livello di padronanza (C2) *Proficient level (C2)*

11) Per quanti anni hai studiato inglese? *For how many years have you been studying English?*

12) Sei in possesso di certificazioni? *Do you have any certificate?*

12.1) Quali? *Which ones?*

13) Hai mai usato chatbot per conversare o imparare una lingua straniera? *Have you ever used a chatbot for chatting or learning a foreign language?*

13.1) Quali? *Which ones?*

Dati sull'interazione *Data on the interaction*

14) Quale chatbot hai utilizzato? *Which chatbot did you use?*

15) Quale dispositivo hai utilizzato? *Which device did you use?*

- Cellulare *Smartphone*
- Computer
- Tablet

16) Hai utilizzato la sintesi vocale? *Have you ever used speech synthesis?*

17) Indica quanto sei d'accordo con le seguenti affermazioni, esprimendo un giudizio da 1 [totale disaccordo] a 5 [totale accordo]. *Rate how much you agree with the following statements on a scale of 1 [total disagreement] to 5 [total agreement].*

- Sono riuscito a capire agevolmente le domande/i messaggi del chatbot.
  I was able to easily understand the chatbot's questions/messages.
- Quando non capivo, il chatbot dava spiegazioni utili e adeguate.
  When I did not understand, the chatbot gave useful and adequate explanations.
- Il dialogo con il chatbot si è svolto in modo coerente.
  The dialogue with the chatbot was coherent.
- Le risposte di feedback da parte del chatbot mi sono sembrate utili (per esempio correzioni, espressioni alternative, brevi valutazioni).
  I found the feedback responses from the chatbot useful (e.g. corrections, alternative expressions, short evaluations).
- Le risposte del chatbot non mi hanno fatto/a sentire discriminato/a o giudicato/a in alcun modo a causa di genere, etnia, abilità o altri fattori.
  Chatbot's answers did not make me feel discriminated against or judged in any way because of my gender, ethnicity, abilities or other factors.
- Mi sono sentito/a coinvolto/a nelle interazioni con il chatbot, come se avessi parlato con una persona.
  I felt involved in the interactions with the chatbot, as if I were talking with a person.
- Durante tutto l'esercizio non ho provato ansia, stress o pressione.
  Throughout the exercise, I did not feel anxious, stressed or under pressure.

- Sono riuscito/a ad utilizzare lo strumento senza bisogno di aiuto esterno.
  I managed to use the tool without needing any external help.
- Mi sono sentito/a stanco/a o annoiato dopo l'interazione con il chatbot.
  I felt tired or bored after interacting with the chatbot.

18) Sei motivato/a continuare a interagire in inglese con il chatbot? *Are you motivated to keep interacting in English with the chatbot?*

19) Hai preferito l'esercizio di dialogo o di role play? *Have you preferred the small talk or role play exercise?*

20) Hai voglia di dirci perché? *Feel like telling us why?*

21) Per finire, se vuoi, commenta gli aspetti positivi e/o negativi della tua interazione col chatbot o condividi con noi qualunque osservazione desideri. *To conclude, if you wish, comment on the positive and/or negative aspects of your interaction with the chatbot or share with us any other observation.*

B. Error Annotation Tags for UNITE

Category	Tag	Description
Digitally-Mediated Communication	<DMCC>	Capitalization issues (i.e., uncapitalized “I”, uncapitalized start of a sentence or turn).
	<DMCA>	Use of abbreviations in digitally-mediated communication (e.g., *OK*, *lol*, etc.).
Form	<FS>	Spelling errors.
	<FM>	Morphological errors involving derivational affixes.
Punctuation	<QC>	Confusion of punctuation marks.
	<QL>	Punctuation mark instead of lexical item (or vice versa).
	<QM>	Missing punctuation.
	<QR>	Redundant punctuation.
Grammar	<GDD>	Errors with demonstrative determiners (e.g., this, that, etc.).
	<GDO>	Errors with possessive determiners (e.g., my, your, etc.).
	<GDI>	Errors with indefinite determiners (e.g., any, some, etc.).
	<GDT>	Errors with other types of determiners (e.g., relative, interrogative, etc.).
	<GA>	Errors with definite, indefinite, or zero articles.
	<GADJCS>	Errors with comparative or superlative adjectives.
	<GADJN>	Errors with adjective number.
	<GADJO>	Errors with adjective order.
	<GADVO>	Misplaced adverbs.
	<GNC>	Errors with noun case (e.g., Saxon genitive misuse).
	<GNN>	Errors with noun number.
	<GPD>	Errors with demonstrative pronouns (e.g., this, that, etc.).
	<GPP>	Errors with personal pronouns (e.g., you, we, etc.).
	<GPO>	Errors with possessive pronouns (e.g., mine, yours, etc.).

	<GPI>	Errors with indefinite pronouns. (e.g. anyone, nothing, etc.).
	<GPF>	Errors with reflexive or reciprocal pronouns (e.g., myself, each other, etc.).
	<GPR>	Errors with relative or interrogative pronouns (e.g., what, which, etc.).
	<GPU>	Unclear pronominal reference.
	<GVAUX>	Misuse of primary, modal, or semi-auxiliaries (e.g., do, have, etc.).
	<GVM>	Errors with verb morphology.
	<GVN>	Errors with subject-verb agreement.
	<GVNF>	Errors in use of *-ing*, infinitives, or relative clauses.
	<GVT>	Misuse of tense or aspect.
	<GVV>	Errors with active/passive voice.
	<GWC>	Errors due to confusion between word classes.
Lexico-Grammar	<XADJCO>	Errors with adjective complementation.
	<XNCO>	Errors with noun complementation.
	<XPRCO>	Errors with preposition complementation.
	<XVCO>	Errors with verb complementation.
	<XADJPR>	Errors with adjective dependent prepositions.
	<XADVPR>	Errors with adverb dependent prepositions.
	<XNPR>	Errors involving dependent prepositions for nouns.
	<XVPR>	Errors involving dependent prepositions for verbs.
	<XNUC>	Errors in the use of uncountable/countable nouns.
Lexis	<LCC>	Errors in the use of coordinating conjunctions (e.g., *and*, *or*, etc.).
	<LCS>	Errors in the use of subordinating conjunctions (e.g., *although*, *because*, etc.).
	<LCLS>	Errors in the use of single logical connectors (e.g., *nevertheless*, etc.).

	<LCLC>	Errors in the use of complex logical connectors (e.g., *on the other hand*, etc.).
	<LSADJ>	Conceptual or collocational errors involving adjectives.
	<LSADV>	Conceptual or collocational errors involving adverbs.
	<LSN>	Conceptual or collocational errors involving nouns.
	<LSPR>	Conceptual or collocational errors involving prepositions.
	<LSV>	Conceptual or collocational errors involving verbs.
	<LP>	Errors in fixed word combinations, including idioms, compounds, and phrasal verbs.
	<LWCO>	Instances of coined words, calques, or invented proper nouns.
Word	<WM>	Missing words.
	<WR>	Redundant words.
	<WO>	Errors in word order.
Infelicities	<Z>	Stylistic problems or unclear sequences requiring reformulation.
Code-Switching	<CS-INTRA>	Code-switching within a sentence.
	<CS-INTER>	Code-switching between sentences or conversational turns.

C. Instructions for the Custom GPT

Context

You are a helpful and friendly annotator specialized in assisting with analyzing and annotating learner corpora of conversations between Italian EFL students and chatbots for linguistic research and educational purposes. You use a comprehensive Error Tagging Manual (`annotation_manual.md`) in markdown format as your primary reference. This manual includes predefined tags, instructions, and examples for annotating errors in learner-produced texts.

Your role is to accurately identify and tag errors in students' conversational turns, using the manual to ensure consistency. You maintain a formal yet supportive tone and clarify ambiguities when user guidelines are unclear or incomplete.

Annotation Process

Trigger:
The user asks to annotate a text file.

Instructions:
1. **Preparation**
 - Always access the provided `annotation_manual.md` in the knowledge section and revise error tags to ensure familiarity.
 - Ask the user if they want you to annotate:
  - Only the first two student turns as a checkpoint.
  - A specific section of the file.
  - The entire file in one go (with an option for the user to review). Default to the first two turns if unclear.
 - If the user gives your further instructions, include them in the annotation process.

2. **Annotation**
 - For each student's turn in the agreed portion of the file break down the process into manageable steps:
  - Apply the guidelines provided in the in next section of these instructions (**"Tagging Guidelines"**).
  - Analyze the content and identify potential issues.
  - Choose the right tags.
  - Verify whether each identified issue constitutes an actual error as per the `annotation_manual.md`.
  - Only tag capitalization errors if they clearly violate the rules in `annotation_manual.md`.
   - **Example of Original:** `Hi how are you?`
   Example of Output: `Hi how are you?` (leave untagged)
   - **Example of Original:** `hi how are you?`
   Example of Output:`<DMCC corr="Hi">hi</DMCC> how are you?`

- Verify where to place the tag where to place the tag according to the instructions in the "Annexes" section of the `annotation_manual.md`.
  - Evaluate the possibility of using nested tags.
  - For unclear errors, flag with a comment: `<TAG corr="unknown">...</TAG> <!--Explanation of ambiguity -->`
- Present the annotated portion to user for review.

3. **User Review**
- Present annotations in a numbered list or another clear format specified by the user.
- Ask the user to confirm or provide feedback. If no feedback is given, proceed with caution.
- Revise tags based on user feedback and confirm changes before finalizing.

4. **Finalization**
- After reaching the end of the file, ask the user if they want you to save in a new file:
  - Both original and annotated content.
  - Only annotated content.
- Ask the user in which format the file should be saved.
- Ask the desired filename.
- Save the file according to the user's specifications.
- Notify the user that the file is ready.

Tagging Guidelines

- Always consult `annotation_manual.md` in the knowledge base to ensure tagging aligns with its definitions, examples, and instructions.
- Avoid applying tags to correct content. If an issue is ambiguous, flag it for user review instead of tagging it.
  - **Example:** `Hi how are you? <!-- No tagging applied: capitalization appears correct -->`
- Never insert missing full stops at the end of a turn.
- Always choose the tag on the basis of the **incorrect word/phrase**.
  - **Example:**
    ```xml
    Do you have other recipes <GA corr="for">a</GA> gluten-free and vegan food?
    ```
    Explanation: The tag `<GA>` (Grammar Article) is used because the issue involves the misuse of an article.
- Never tag errors caused by your own annotations, **except** when correcting a verb tense leads to additional verb tense errors in the same sentence.
  - **Original:**
    ```xml
    I always liked anime and manga, so I found this as an opportunity to understand them in their original language.
    ```

Output - Step 1:
```xml
I <GVT corr="have always liked">always liked</GVT> anime and manga, so I found this as an opportunity to understand them in their original language.
```
Output - Step 2:
```xml
I <GVT corr="have always liked">always liked</GVT> anime and manga, so I <GVT corr="find">found</GVT> this as an opportunity to understand them in their original language.
```

Error Explanation Requests

- Explain applied tags with examples and supportive reasoning from the manual.
- If the user disagrees with an annotation, revise the tags accordingly and ask for feedback to avoid future misinterpretations.

D. Prompts for the Custom GPT

Prompt for initializing the conversation:

Can you help me annotate this text?

Prompt for the annotation task:

Start with the first *N* turns and show them in XML format.
//
Annotate the entire document and show it in XML format.

Example:
```xml
1. Original: I woul like to tolck about my travel in Malta
   Annotated: I <FS corr="would">woul</FS> like to <FS corr="talk">tolck</FS> about my <LSN corr="trip">travel</LSN> <LSPR corr="to">in</LSPR> Malta

2. Original: i would like only specking about you
   Annotated: <DMCC corr="I">i</DMCC> <XVCO corr="would like only to speak">would like only <WM corr="to">\0</WM> <FS corr="speaking">specking</FS></XVCO> about you
```

While annotating pay attention to:
- Choose the tag on the basis of the incorrect element.
- Where to correctly place the tag.
- How to tag capitalization issues.
- How to handle missing or redundant elements.
- The possibility of using nested tags for multiple errors in the same word/phrase.
- Avoid adding a full stop at the end unless it is present in the original text.

Take your time to analyze each sentence thoroughly.

Follow-up prompt (for step-by-step annotation):

This approach looks good! Continue by annotating the following *N* turns.
While annotating pay attention to:
- Choose the tag on the basis of the incorrect element.
- Where to correctly place the tag.
- How to tag capitalization issues.
- How to handle missing or redundant elements.
- The possibility of using nested tags for multiple errors in the same word/phrase.
- Avoid adding a full stop at the end unless it is present in the original text.
Take your time to analyze each sentence thoroughly.

Prompt for saving the annotation:

I would like to save the annotation in TXT format using UTF-8 encoding:
- Only include the annotated version of each turn, even those where no correction was made.
- Separate each turn using `\n\n`.
- Use the literal string `\0` (backslash zero) for missing words or deletions, not the NUL character (`\x00`).
- Use `<` and `>` for tags, not `<` and `>`.
- Append `_gpt` to the original filename (e.g., `example.txt` → `example_gpt.txt`).