Master Thesis

Grammatical Error Detection in L2 English and Italian: How Multilingual LLMs Handle Ambiguity in Learner Errors

Elisabetta Dentico

Supervisor Luís Morgado da Costa
2^{nd} reader Isa Maks

a thesis submitted in fulfillment of the requirements for the degree of

MA Linguistics
(Text Mining)

Vrije Universiteit Amsterdam

Computational Linguistics and Text-Mining Lab
Department of Language and Communication
Faculty of Humanities

Date	August 15, 2025
Student number	2843401
Word count	37585

Abstract

This thesis explores how large language models deal with ambiguity in grammatical errors in learner-written English and Italian. Using data from the MultiGED-2023 shared task and learner corpora (MERLIN, FCE, REALEC), five transformer-based models per language, including monolingual and multilingual variants, were fine-tuned and evaluated on token-level classification tasks.

In addition to comparing model performance across architectures and languages, the study examines how predictions diverge from gold labels, and whether such divergences reflect valid linguistic alternatives for error detection/correction.

A qualitative error analysis reveals patterns linking model output to specific grammatical error types, highlighting areas where models struggle or produce overgeneralized corrections. These findings underscore the need for more flexible evaluation methods and contribute to a deeper understanding of ambiguity and variation in L2 grammar detection.

A generative model, QWEN, was also tested for its ability to produce plausible corrections beyond the scope of traditional GED tasks. By generating diverse outputs, the model provided insight into the range of linguistically acceptable revisions and helped assess whether prediction errors were genuinely incorrect or simply unannotated variants.

Declaration of Authorship

I, author, declare that this thesis, titled *Grammatical Error Detection in L2 English and Italian: How Multilingual LLMs Handle Ambiguity in Learner Errors* and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a degree degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date: August 15, 2025

Signed: 

Acknowledgments

I would like to express my sincere gratitude to my supervisor, Luís, for his support, valuable insights, and expert guidance throughout this research journey. His systematic and precise approach to supervision ensured that I never felt alone in tackling the challenges of this work. His dedication and mentorship over the past months have been crucial in exploring a topic that holds such personal significance and importance to me.

I would also like to thank Isa for serving as my second reader and for taking the time to review this work.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

In recent years, grammatical error detection (GED) has become increasingly relevant for both language learning and assessment, particularly in multilingual contexts. GED systems aim to identify ungrammatical elements in learner-produced text and are widely used in educational NLP applications such as automated feedback, error correction, and writing support (Lee and Lee, 2013). However, the task remains complex due to the ambiguous and variable nature of learner language, especially when multiple plausible corrections are possible or when deviations reflect acceptable non-standard usage. These challenges are particularly pronounced in under-resourced languages, where less annotated data is available.

This thesis investigates how fine-tuned monolingual and multilingual large language models, particularly BERT-based architectures, handle grammatical errors in learner texts written in Italian and English as second languages. The study is grounded in the context of the MultiGED 2023 shared task, which provides a multilingual benchmark for GED. The dataset used in this task is derived and adapted from the MERLIN corpus (Boyd et al., 2014) for Italian, and from the FCE (Yannakoudakis et al., 2011) and REALEC (Kuzmenko and Kutuzov, 2014) corpora for English. These corpora present authentic learner productions and are annotated with token-level labels. While the shared task simplifies the evaluation format to a binary classification task—labeling each token as either correct or incorrect, this study goes beyond by conducting a fine-grained error analysis that takes into account multiple levels of ambiguity and annotation consistency.

The central objective is to assess how well these fine-tuned models capture error patterns in learner data and how they respond to syntactic ambiguity and multiple correction possibilities. In particular, this thesis examines situations where model predictions diverge from gold-standard labels, evaluating whether such divergences reflect nonsensical outputs or linguistically valid alternatives that are not annotated. In doing so, the study aims to better understand the limitations of current evaluation metrics and highlight the need for more flexible assessment frameworks in GED research.

To guide this investigation, the thesis addresses the following research questions:

1. How do performances of different fine-tuned encoder architectures compare to one another?

2. Are there consistent patterns in model predictions that can be mapped to specific grammatical error types, and what do these patterns reveal about model strengths or limitations?

3. When a prediction deviates from the expected outcome, is there evidence that it aligns with alternative types of corrections?

4. Can generative language models be leveraged to generate corrections, broadening the evaluation framework for this task?

These questions reflects a broader aim of the thesis: to move beyond the token-level binary evaluation of the MultiGED-2023 shared task and toward a linguistically grounded analysis of model behavior.

To achieve this, five transformer-based models are fine-tuned for each language, including two monolingual models (DistilBERT and either BERT-base-cased or Italian-BERT in both base and XXL variants), two variants of multilingual BERT (cased and uncased), and one multilingual model based on XLM-RoBERTa. These models are evaluated on the respective English and Italian subsets of the MultiGED dataset. To be in line with findings from top-performing teams in the shared task, the models are trained separately for each language to avoid cross-lingual interference, given the significant morphological and syntactic divergence between English and Italian.

In the second part of the study, a qualitative error analysis is performed to complement the quantitative results. This analysis leverages richer annotations from the original corpora and maps the model outputs to the extracted error types, comparing both to a comprehensive human-generated correction and to up to five generated corrections, which are manually assessed for their validity and alignment with human judgments. This dual approach allows for a more nuanced evaluation of model behavior in ambiguous contexts.

By combining classification-based modeling with generative evaluation and in-depth error analysis, this research contributes to our understanding of how multilingual LLMs handle ambiguity in GED tasks. The findings have broader implications for educational NLP, particularly in the design of more robust and interpretable systems for grammar correction and language learning support in both high-resource and low-resource settings.

All code and experimental materials used in this study are available at: `https://github.com/elisabetta1999/Final-Thesis-MultiGED-2023/tree/main`.

Chapter 2

Background

This chapter provides a structured overview of grammatical error detection (GED) in natural language processing, going through its key concepts, methodologies, and recent advances. It begins by defining grammatical errors and examining the types of mistakes typically made by language learners. The chapter then traces the evolution of GED approaches, from traditional rule-based and statistical methods to modern neural and transformer-based models, highlighting how these advances have improved detection accuracy. Additionally, it reviews the role of shared tasks and datasets in shaping research, with a particular focus on the shift from English-centric to multilingual GED efforts. A dedicated section discusses the datasets used for this work, including their characteristics and relevance, as well as the models employed, both discriminative models, which focus on token-level classification, and generative models, which produce corrected versions based on the detected errors.

2.1 What is a *grammatical error*?

Before defining grammatical error detection, it is important to clarify what typically constitutes a grammatical error in grammatical error detection tasks. The term is not limited to violations of morphological and syntactic rules specific to a language, but is also used to refer to pragmatic misuses of language, meaning the use of inappropriate forms in a given communicative context (Leacock et al., 2014). This broader definition includes difficulties with prepositions, punctuation, and even lexical choices that are inappropriate for a particular topic or required register.

While this study does not adopt a single, fixed taxonomy of error types, due to the need to adapt to the classification systems used in each of the datasets considered, the classification proposed by Garofolin et al. (2016) offers a useful reference point. Its typology provides a linguistically grounded and structured framework that aligns closely with the annotation schemes adopted across the datasets used in this research. Specifically, it distinguish between:

(a) phonological errors, including spelling errors;

(b) morphological errors, involving inflectional mistakes in nouns, verbs, articles, and prepositions;

(c) morphosyntactic errors, defined as legitimate forms that are nonetheless inappropriate within the context, that is, the syntactic and grammatical relations that link linguistic units within an utterance;

(d) syntactic errors, such as word order issues or omission of function words;

(e) lexical errors, including word choice problems, invented words, or L1 interference;
(f) stylistic errors, which affect cohesion, coherence, and punctuation.

Although spelling errors are not usually classified as grammatical, they can affect the grammaticality of a sentence by obscuring correct morphological forms. Such errors may result from the misapplication of morphological rules. For example, in English, learners might confuse homophones such as 'their' and 'they're', writing 'Their going to school' instead of 'They're going to school'. While this is technically a spelling error, it obscures the contracted verb form 'they are', affecting the morphological and syntactic structure of the sentence. In Italian, learners may incorrectly use indefinite articles, such as the feminine article 'un" instead of the masculine 'un' before a noun beginning with a vowel (e.g., 'un' uomo' instead of 'un uomo'), which disrupts grammatical agreement.

2.2 Grammatical Error Detection: Related Work

Grammatical error detection (GED) refers to the task of identifying whether one or more tokens in a learner's text are ungrammatical, often labeling them without necessarily suggesting corrections. Grammatical error correction (GEC), by contrast, involves an initial detection step which leads to automatically generating corrected versions of such errors. While GEC is often more prominent in end-user applications, GED plays a crucial role as a preliminary step, particularly for educational feedback and error analysis.

The main types of approach of detecting grammatical errors within a sentence can be classified into two categories (Lee and Lee, 2013):

1. rule-based approaches;

2. data-driven approaches.

2.2.1 Rule-Based Approaches

Rule-based approaches represent the earliest stage of automated error detection. As the name suggests, this method relies on the use of an extensive set of grammar rules against which learner-produced sentences are compared.

While conceptually straightforward, this approach presents several challenges. It is both time-consuming and complex to elicit and define the full set of rules needed to adequately detect errors. Moreover, deciding which rules should be included is not always clear-cut. Some grammatical rules may only apply to specific registers, whereas informal or colloquial language often allows for more flexible and diverse structures.

Like any approach, rule-based systems have their strengths and weaknesses. A key strength lies in the high coverage that explicit rules can offer, especially in controlled contexts. However, this is counterbalanced by serious scalability issues and limited flexibility, particularly when dealing with idiomatic expressions or non-standard language.

Before the advent of statistical methods, error detection tools relied on parsers based on large, manually written computational grammars. These grammars, developed by linguists, needed to be *error-tolerant*, therefore capable of parsing sentences and identifying syntactic relationships even in the presence of grammatical mistakes (Leacock et al., 2014).

The main difficulty in applying such parsers to L2 texts lies in the fact that learner language typically contains far more errors than native (L1) texts. This high error

density can significantly affect the parser's performance, making it difficult to analyze the sentence correctly or offer meaningful feedback.

2.2.2 Data-Driven Approaches

As highlighted by (Leacock et al., 2014), data-driven approaches help mitigate the issue of error intolerance, a limitation that often hinders the effectiveness of rule-based systems. However, these approaches typically require large amounts of annotated data, which can be both costly and time-consuming to produce, and may also contain annotation errors. To address this, some methods aim to reduce or eliminate the reliance on manually annotated data, as demonstrated in the work of Yarowsky (1994) and Golding (1996).

Within the data-driven paradigm, two main types of systems can be identified:

1. Statistical ML
2. Neural/Deep Learning

Statistical systems, particularly parsers that rely on probabilistic methods, assign probabilities to each word in a sequence. Based on patterns found in the training data, they may assign lower probabilities to word combinations that were less frequently observed. These sequences might be rare but grammatically correct, or they might represent actual errors. This approach often involves substantial feature engineering to guide the model toward accurate predictions, using statistical models such as Logistic Regression (Burstein et al., 2004) or Naive Bayes (Golding, 1996).

In such systems, the text is analyzed using features that are specifically relevant to the type of error being targeted. For example, the countability of a noun is a key feature for identifying article errors but is generally not useful for detecting preposition errors. Early work in this area used decision trees, but maximum entropy classifiers have since become more wide-spread. Common approaches involve considering as features context n-grams of different sizes, usually within a window of ± 3 to ± 7 tokens, surrounding a potential error (Rei and Yannakoudakis, 2016). Additional features may include part-of-speech tags and dependency paths.

The approaches mentioned above generally target specific error types, which means that the features extracted are tailored to particular error classes. This focus enables fine-grained rule-based or statistical detection, but it also limits generalizability. As the field moved toward broader grammatical error detection (i.e., systems capable of addressing a wide range of error types), feature engineering became a bottleneck. To overcome this limitation, recent research has increasingly turned to neural architectures that bypass the need for manually designed features (Rei and Yannakoudakis, 2016).

Within the neural/deep learning approach to Grammatical Error Detection (GED), two methodological stages can be identified, reflecting the evolution of the field alongside technological advances. Early neural approaches relied on sequence labeling models such as Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) networks, and Bidirectional LSTM (BiLSTM) models (Rei and Yannakoudakis, 2016). These models take a sequence of tokens as input and output a probability for each token indicating whether it is grammatically correct or incorrect. Each token is mapped to a vector representation, which is updated through composition functions (e.g., recurrent layers). The resulting output vectors are passed through a softmax layer, which converts them into a probability distribution over possible output classes, assigning each

class a value between 0 and 1 that sums to 1 (Singh, 2023). Rei and Yannakoudakis (2016), in particular, explored the performance of six different neural architectures on a GED task, including multi-layer variants of bidirectional LSTM, bidirectional recurrent and convolutional neural architectures. However, this approach has shown inconsistent performance on out-of-domain data (Bell et al., 2019).

With the advent of deep learning, contextualized word embeddings became a crucial element in GED systems. Bell et al. (2019) evaluated the effectiveness of integrating BERT (Devlin et al., 2019), ELMo (Peters et al., 2018), and Flair (Akbik et al., 2018) embeddings into a BiLSTM sequence labeling architecture. Their model outperformed previous systems and generalized well to out-of-domain data. Notably, they found that missing word errors were the most difficult to detect.

Despite such promising performance, LSTM and BiLSTM architectures still faced a key limitation: the scarcity of error-annotated training data. To address this, researchers relied on artificially generated datasets, created by introducing grammatical errors into grammatically correct sentences. These synthetic corpora were crucial for training neural sequence labeling models effectively (Rei and Yannakoudakis, 2017; Kasewa et al., 2018).

This dependence on synthetic data has been partially mitigated by the emergence of transformer-based architectures, as these models are pre-trained on large and diverse corpora, enabling them to acquire broad linguistic knowledge that can be effectively adapted to a wide range of tasks. Since around 2018, transformers have become the dominant approach in GED, largely replacing earlier neural models. Their ability to leverage large-scale pretraining on raw text enables them to capture rich contextual information without the need for handcrafted features or artificial data. Kaneko and Komachi (2019) demonstrated the effectiveness of using pretrained language models for GED. Building on this, Yuan et al. (2021) fine-tuned multiple transformer-based models, BERT, XLNet (Yang et al., 2019), and ELECTRA (Clark et al., 2020), achieving state-of-the-art results on binary GED evaluation across several benchmark datasets, including FCE (Yannakoudakis et al., 2011), BEA(Bryant et al., 2019), and CoNLL-2014(Ng et al., 2014), which will be discussed in more detail in the following paragraph.

2.2.3 Shared Tasks for Grammatical Error Detection

The evolution of GED methodologies, alongside advancements in technology and model architectures, is particularly evident when examining the progression of shared tasks in the field. These tasks have not only encouraged increasingly competitive approaches to GED, but their evolving design also reflects broader developments in the field: from the nature of the data provided, to the goals of the tasks, and the strategies adopted by participants to address the problem. Even though the following shared tasks are mainly focused on grammatical error correction rather than detection, which, of course, requires a preliminary search of the grammatical errors within the text, some of them have 'detection' as one of the evaluation metrics.

One of the first approaches to grammatical error detection (GED) was the HOO 2011 shared task (Dale and Kilgarriff, 2011), which targeted a broad set of error types, making the task significantly complex. Due to this complexity, the goal was redefined the following year with the proposal of the HOO 2012 shared task (Zesch and Haase, 2012). In this task, fourteen participating teams were challenged to detect, recognize, and correct errors related to article and preposition usage in English, two categories identified by the organizers as particularly problematic for English learners.

The training, development, and test sets were drawn from the Cambridge Learner Corpus, specifically using FCE texts from 2000 and 2001 (Yannakoudakis et al., 2011). To address the task, teams employed a mix of rule-based and statistical approaches (Bhaskar et al., 2012; Dahlmeier et al., 2012).

The 2013 CoNLL shared task (Ng et al., 2013) focused on five error types, articles, prepositions, noun number, verb form, and subject-verb agreement, within a broader grammatical error correction task. The training data consisted of an adapted version of the NUCLE corpus (Dahlmeier and Ng, 2013), which contains essays written by students of the National University of Singapore. Early shared tasks like CoNLL-2013 primarily used classifiers trained separately for each error category. The most common methodological approaches included maximum entropy classifiers, valued for their flexibility in incorporating diverse linguistic features, together with rule-based systems often employed independently or in hybrid pipelines.

Only one year later, the CoNLL-2014 shared task (Ng et al., 2014) introduced several important changes. The number of error types to identify increased from 5 to 28, while using the same dataset, reflecting the rise of hybrid systems capable of detecting multiple error types simultaneously. The evaluation metric shifted from the previously used F1-score to the F0.5 score, emphasizing precision over recall by weighting the former twice as much as the latter (Gaurav, 2023). This evaluation metric, which will also be adopted in this thesis, has become standard in GED and GEC systems starting from this specific shared task. Approaches often combined rule-based methods, n-gram language models, and statistical machine translation (SMT) techniques. Compared to the previous year, fewer machine-learned classifiers were used.

Lastly, the BEA-2019 shared task (Bryant et al., 2019) took place after a gap of about five years without a major shared task in grammatical error detection. Like earlier tasks, GED was considered only as a subtask within a broader grammatical error correction challenge. This shared task aimed to renew attention on GED, especially given the advances in neural approaches during the intervening years. To support this, a new dataset was introduced: the Cambridge English Write & Improve (W&I) (Cambridge University Press and University of Cambridge, n.d.) and LOCNESS (Granger, 2014) corpora. While both corpora include essays from learners around the world, LOCNESS also contains essays written by native American and British English speakers. As in CoNLL-2014, detection performance was evaluated using the F0.5 score. Teams mainly used neural models, especially transformer-based neural machine translation (NMT) ensembles. Pretrained transformers like BERT and GPT-2 were used for error detection. Some systems also combined CNN encoder-decoders with spellcheckers.

All the aforementioned shared tasks treat GED merely as a preliminary step toward grammatical error correction. Research attention to GED itself remains marginal, especially for languages other than English. This is why I chose to participate in the Multilingual Grammatical Error Detection shared task. Moreover, this shared task gives me the opportunity to explore not only the state-of-the-art for English but also for my native language, Italian, which has not yet been deeply and thoroughly evaluated in grammatical error detection.

2.3 The Multilingual Grammatical Error Detection Shared Task

The 2023 Multilingual Grammatical Error Detection (MultiGED-2023) shared task (Volodina et al., 2023) is the first initiative organized by the *Computational SLA* working group to address the need for implementing grammatical error detection across multiple, less-resourced languages. Specifically, it includes five languages: Czech, English, German, Italian, and Swedish.

Historically, the development of GED has been heavily centered on English, primarily because English is the most widely learned second language worldwide (Leacock et al., 2014). This focus has significantly shaped the progress and applicability of NLP and Computer-Assisted Language Learning (CALL) tools for other languages. A major limiting factor in expanding GED to other languages is the lack of annotated learner data. Despite the evident difficulty in accessing equal and balanced amounts of annotated data for each of the languages, the decision to use only L2 original data is grounded in the conviction expressed by Leacock et al. (2014), who argued that error correction tools benefit greatly from being trained on authentic texts written by L2 learners.

The overrepresentation of English in GED and GEC research has historically excluded other languages. A multilingual shared task is therefore essential, particularly because these languages do not yet have sufficient annotated data to support dedicated monolingual shared tasks. To mitigate this limitation, the organizers allowed data augmentation, cross-lingual coordination, domain adaptation, and transfer learning techniques. For this reason, the task was designed as an open track: given the limited amount of available data, participating teams were permitted to enhance the training data for each language however they preferred, provided they shared their results and techniques with the research community.

Six different teams from five countries (China, Italy, Norway, Sweden, and Vietnam) participated in the competition. One team developed a system exclusively for a single language (Swedish), while the others submitted systems covering all five languages. As previously mentioned, my study focuses specifically on English and Italian.

2.3.1 Evaluation and Results

The performance of each team in the shared task was evaluated using the token-based F0.5 score. Among all participating teams, three consistently demonstrated high performance across all languages, sharing the podium: EliCoDe (Colla et al., 2023), DSL-MIM-HUS (Le-Hong et al., 2023), and Brainstorm Thinkers (Volodina et al., 2023). The results for Italian were notably higher compared to English. The top-performing team, EliCoDe, achieved an F0.5 score of 82.15 for Italian, whereas the scores for English datasets FCE and REALEC were 67.40, DSL-MIM-HUS, and 50.86, EliCoDe, respectively. According to Volodina et al. (2023), this disparity can be attributed to the higher quality and consistency of error annotations in the MERLIN dataset.

The overall best-performing team was EliCoDe, which achieved the highest scores for all datasets except for English REALEC, where DSL-MIM-HUS ranked first. All three top-performing teams used BERT-based models: EliCoDe and DSL-MIM-HUS[1]

[1]The team developed two neural models: a supervised LSTM trained on a character-based representation, and a pretrained BERT model fine-tuned on a subword-token representation (Le-Hong et al.,

employed XLM-RoBERTa, while Brainstorm Thinkers used Multilingual BERT (mBERT) (Devlin et al., 2019).

However, the main difference among the teams lies in how they handled the multilingual data. EliCoDe trained five separate models, one for each language, starting with training on the training set alone, followed by retraining on the combination of training and development sets. The team reported that multilingual training did not lead to improvements, likely due to significant differences between languages in the shared task (Colla et al., 2023). Brainstorm Thinkers adopted a similar approach, training separate models for each language, though specific details about their data handling were not disclosed.

In contrast, DSL-MIM-HUS trained a single multilingual model using XLM-RoBERTa. They combined all datasets into a single corpus, which was then randomly split into training, development, and test sets in proportions of 80%, 10%, and 10%, respectively (Le-Hong et al., 2023).

2.3.2 Shared Task Data

Unlike many previous shared tasks or GED research efforts, MultiGED-2023 simplifies the classification task by using only binary labels, *i* for incorrect and *c* for correct, instead of categorizing errors by type. Tokens following a missing word (i.e., omission errors) are also labeled as incorrect to accurately capture this error type. These decisions promote generalizability, as they avoid the complications that arise from differing annotation schemes in the original L2 datasets across languages. However, they can also be seen as a limitation. This point will be further discussed in the Discussion. The dataset used in the shared task is derived from various previously annotated corpora, and considerable effort was made to create a unified classification framework across different languages.

For English and Italian specifically, three datasets were used: two for English and one for Italian.

The English datasets comprise the FCE, *First Certificate in English* (Yannakoudakis et al., 2011), and REALEC, *Russian Error-Annotated Learner English Corpus* (Kuzmenko and Kutuzov, 2014), corpora, while the Italian data come from the MERLIN corpus (Boyd et al., 2014). A comprehensive overview of these datasets is provided in Table 2.1, which summarizes the number of sentences, tokens, annotated errors (i.e., tokens labeled as incorrect), and error rates for each corpus split. Notably, the Italian MERLIN corpus shows a higher overall error rate compared to the English datasets. The dataset statistics were extracted from publicly available resources on GitHub (Språkbanken, 2023). While the training and development sets include gold labels, as required for supervised classification, the test sets for both languages do not contain any gold label information. Apart from the aggregate statistics shown in Table 2.1, no further information about the test sets is available. This is standard practice in shared tasks to ensure an unbiased evaluation of model performance.

2023).

Dataset	Split	Sentences	Tokens	Errors	Error Rate
FCE	Train	28,357	454,736	42,899	9.4%
FCE	Development	2,191	34,748	3,460	10.0%
FCE	Test	2,695	41,932	4,501	10.7%
FCE	Total	33,243	531,416	50,860	9.6%
REALEC	Development	4,067	88,008	8,103	9.2%
REALEC	Test	4,069	89,761	8,505	9.5%
REALEC	Total	8,136	177,769	16,608	9.3%
MERLIN	Train	6,394	80,336	12,190	15.2%
MERLIN	Development	758	9,144	1,211	13.2%
MERLIN	Test	797	10,218	1,492	14.6%
MERLIN	Total	7,949	99,698	14,893	14.9%

Table 2.1: Summary of dataset statistics for English (FCE, REALEC) and Italian (MERLIN) learner corpora, including sentences, tokens, annotated errors, and error rates.

2.4 Original Datasets

The datasets described above were derived from the original corpora through a series of transformations, including preprocessing steps such as tokenization and the removal of error labels. In the second part of this thesis, I will also make use of the original datasets in order to map the model's predictions to specific error types and investigate whether the model systematically struggles with certain categories of errors. The following paragraphs describe the structure of the original datasets, the way error types are presented, and the linguistic phenomena they aim to capture.

2.4.1 FCE

The FCE corpus is a subset of the larger Cambridge Learner Corpus (CLC) (Nicholls, 2003), which comprises scripts from ESOL (English for Speakers of Other Languages) examinations. Specifically, the FCE dataset includes only the texts produced during the First Certificate in English, corresponding to CEFR level B2 (Yannakoudakis et al., 2011).

The scripts are first anonymized and then annotated using an XML structure. Each XML file contains not only the exam text and learner responses, but also metadata, personal information about the learner such as grade, age, and native language. The tasks assigned to learners include short texts such as reports, letters, or short stories, typically no longer than 400 words. Each script is also tagged with approximately 80 different error types, providing rich annotation for error analysis.

To ensure annotation consistency, the CLC dataset has been manually annotated by two coders. The annotation includes not only the error type tag but also a suggested correction, provided only when the correct form is clear, certain, and direct, without any attempt to paraphrase or alter the learner's intended style. Errors are represented in the following format:

```
Thanks for <NS type="DD"><i>you</i><c>your</c></NS>letter.
```

The incorrect token is enclosed within the `<i>` tag, the correction is provided within the `<c>` tag, and the entire correction pair is wrapped in an `<NS>` tag with an error type attribute (in this case, `DD` for "wrongly Derived Determiner").

Most error labels follow a two-letter coding system, where the first letter denotes the general type of error (e.g., substitution, omission) as displayed in table 2.2, and the second indicates the word class of the affected token (Nicholls, 2003, 573–574), as visible in table 2.3:

General types of error (first letter)

Code	Description
F	Wrong Form Used
M	Something Missing
R	Word or Phrase needs Replacing
U	Word or Phrase is Unnecessary (i.e. redundant)
D	Word is wrongly Derived

Table 2.2: General types of error codes used in the Cambridge Learner Corpus

Word classes (second letter)

Code	Description
A	Pronoun (Anaphoric)
C	Conjunction (linking word)
D	Determiner
J	Adjective
N	Noun
Q	Quantifier
T	Preposition
V	Verb (includes modals)
Y	Adverb (-ly)

Table 2.3: Word class codes used in the Cambridge Learner Corpus

In addition to the error types listed in table 2.2 and 2.3, the annotation system also accounts for other common learner errors. These include punctuation errors, which are marked with the letter P as the second element of the code (e.g., MP, UP, etc.); countability errors, where an uncountable noun is mistakenly treated as countable (coded as C + word class); and agreement errors, where grammatical agreement between elements such as subject and verb or determiner and noun is violated (e.g., AG + word class). Another category includes false friends, words that resemble those in the learner's native language but differ in meaning or usage, coded as FF + word class.

When a segment contains more than one error, the codes can be nested, forming a hierarchical annotation structure.

2.4.2 REALEC

The REALEC corpus (Kuzmenko and Kutuzov, 2014) is a large collection of learner essays; however, unlike the FCE corpus, it does not include many manually annotated texts. It contains 18,700 essays written by students from the Higher School

of Economics (HSE) who took the Independent English Language Test (IELTS) between 2014 and 2020. Due to the extensive workload and the continuous maintenance required for such a large dataset, only the texts from 2014 to 2019 were manually annotated. This task was carried out by undergraduate students in Linguistics who demonstrated a proficient level of English. Starting from 2020, manual annotation was replaced with automatic annotation using neural networks, specifically a BERT-based model (BERT-transformer type) for both error detection and correction. For this reason, the MultiGED-2023 shared task considered only the data up to 2019 (Volodina et al., 2023).

The dataset is presented in the `.brat` format, which means that the text and the annotations are stored in separate files. The essay written by the learner appears in a `.txt` file, the annotations are stored in a corresponding `.ann` file, and the metadata, i.e., information about the learner, is stored in a `.json` file.

Errors are represented as follows. Each error is assigned an event ID which appears in the first column of the `.ann` file. This ID begins with "T" followed by a numerical value where "T" stands for text-bound annotation. The three files corresponding to each essay share the same filename (excluding the suffix), allowing them to be matched by naming conventions. In the `.ann` file, the ID is followed by a tab character, then the error type, the character span (index), and the erroneous string. A commented line following the error contains the correction associated with that specific event ID.

```
 T1 Articles 97 108 whole world
#1 AnnotatorNotes T1 the whole world
```

In order to correctly label the errors, annotators were provided with a set of 151 grammar rules, which were used to define six broad categories of errors (Kuzmenko and Kutuzov, 2014):

- punctuation;
- spelling;
- capitalization;
- grammar;
- vocabulary;
- discourse.

These high-level categories are used to indicate the general nature of the error (Vinogradova and Lyashevskaya, 2022). The first three categories, punctuation, spelling, and capitalization, do not include any further internal subdivisions. In contrast, the grammar, vocabulary, and discourse categories allow for more fine-grained classifications.

Within the *grammar* category, tags indicate the part of speech affected by the error, such as *Determiners* (which includes the tag *Articles*), *Verbs*, *Nouns*, and others. Each of these tags may be subdivided further. For instance, the *Verbs* class contains the largest number of subcategories, including tags such as *Tense*, *Modals*, *Voice*, and others. This group also includes syntactic errors, covering aspects such as *Agreement*, *Word order*, *Relative clauses*, and *Confusion of structures*.

Vocabulary errors include mistakes in word choice, lexical selection, or in the formation of derived words, which are labeled as *Derivational affix* errors.

Finally, the *discourse* category refers to errors related to the stylistic level of the text, including issues of coherence or inappropriate register for the intended communicative purpose.

2.4.3 MERLIN

The MERLIN Corpus includes standardized CEFR tests for Italian, German, and Czech. The texts for Italian range from A1 to B2 on the CEFR scale. The data is represented using the EXMARaLDA format, which, unlike other corpora, presents the learner texts in a tokenized structure. The `.exb` files include:

- the tokenized text;
- the lemmas of the tokens;
- the tokenized corrections;
- two mandatory levels of error annotation, with an optional third: the linguistic field and its sublevels (Boyd et al., 2014).

Because multiple interpretations of an utterance are often possible, the annotation process involves collecting and evaluating several options for error classification and correction. This is done to ensure high-quality annotation, ultimately reaching inter-annotator agreement.

Unlike other corpora, the MERLIN corpus includes two types of target hypotheses (TH) (Reznicek et al., 2013):

- TH1, the minimal target hypothesis, provides a correction that remains as close as possible to the surface structure of the learner's utterance;
- TH2, the extended target hypothesis, reflects the intended meaning behind the learner's utterance, making broader changes if necessary.

The minimal target hypothesis, similar to the one used in the Falko essay corpus (Reznicek et al., 2013), consists of a corrected version that differs minimally from the learner text, and forms a grammatical sentence, even if it ignores issues related to semantics, pragmatics, or style.

Not all sentences in the corpus are annotated with both correction tiers, TH1 and TH2. For the MultiGED-2023 shared task, only the minimal target hypothesis (TH1) was used to derive binary error labels. While this simplifies the task setup, it also represents a limitation of the shared task design: relying solely on TH1 not only excludes more natural or fluent alternatives but also forces models into a stricter evaluation regime, where even acceptable variations may be penalized. In the later stages of this thesis, I will incorporate TH2 to investigate how access to alternative corrections affects model performance and error type analysis.

With respect to word order, TH2 is more flexible than TH1. While TH1 follows clear grammatical rules, TH2 allows reordering based on the intended meaning.

In the error annotation process, annotators refer to TH1 when labeling orthographic and grammatical errors, and to TH2 when labeling errors in vocabulary, coherence/cohesion, sociolinguistic appropriateness, and pragmatics.

The most frequent labels used for annotation in the dataset are summarized in the following tables:

Tag	Description
G_Prep	Incorrect use or omission of a preposition
G_Conj	Incorrect use of conjunctions
G_Verb	General verb errors (tense, aspect, mood, etc.)
G_Verb_compl	Errors in verb complementation
G_Inflect_Inexist	Use of a non-existent inflected form
G_Clit	Errors involving clitic pronouns
G_Valency	Valency errors (wrong argument structure or missing complements)
G_Art	Incorrect or missing articles
G_Morphol_Wrong	Morphological errors (e.g. incorrect inflection, gender, number)
O_Punct	Punctuation errors
O_Graph	Graphical/orthographic errors (spelling, character-level issues)

Table 2.4: Linguistic Error Tags in the MERLIN Corpus (markable_scheme)

Code	Description
CHA	Change: the learner's word is replaced with a corrected word
INS	Insertion: a word is inserted in the correction
DEL	Deletion: a word is removed from the learner's original sentence

Table 2.5: Edit Operations in the MERLIN Corpus (TH1Diff and TH2Diff, if applicable)

2.5 Models

This section presents the models used in this thesis, divided into two parts based on their learning objectives: discriminative and generative models. Discriminative models are used in the first stage to perform token-level classification, identifying grammatical errors in learner texts. These models are trained to distinguish between correct and incorrect tokens by learning decision boundaries between classes. In the second stage, a generative model is used to produce corrected versions of the identified errors. Generative models are trained to generate coherent and contextually appropriate output. This two-part setup enables a pipeline in which classification informs generation, combining the strengths of both approaches.

2.5.1 Discriminative Models

Fine-Tuning Pretrained Models

Transformer-based language models such as BERT and RoBERTa are typically pretrained on large, general-purpose corpora using self-supervised objectives, most notably masked language modeling (MLM). In masked language modeling (MLM), the model learns to predict missing or hidden words in a sentence based on their surrounding context. This objective encourages the model to develop a deep understanding of language structure and meaning by leveraging information from both directions in the input (Devlin et al., 2019).

BERT also incorporates a second pretraining objective: next sentence prediction (NSP), designed to help the model learn sentence-level relationships. For this task, the model is presented with pairs of sentences and must predict whether the second sentence logically follows the first (IsNext) or is randomly sampled from the corpus (NotNext). This additional objective was shown to be beneficial for tasks involving inter-sentential reasoning, such as natural language inference and question answering.

In contrast, RoBERTa (Liu et al., 2019) modifies the pretraining setup by removing the NSP objective and instead focusing only on MLM. It also introduces optimizations such as dynamic masking (applying different masks at each epoch), larger mini-batches, and training on significantly more data. These changes were found to improve performance across a variety of NLP benchmarks.

While pretraining equips these models with general-purpose linguistic knowledge, adapting them to specific downstream tasks requires an additional phase known as fine-tuning. Fine-tuning involves continuing the training on a smaller, task-specific dataset with supervised labels. This process adjusts the model's parameters to the nuances of the target task while still leveraging the representational power gained during pretraining. Crucially, fine-tuning is both data- and compute-efficient, making it practical for a wide range of applications including GED.

Overview and Model Selection

To contribute meaningfully to the MultiGED shared task, this study explores the performance of multiple transformer-based models fine-tuned on the MultiGED training data. Only bidirectional encoder architectures were considered, both to align with the experimental setup adopted by the top-performing teams in the shared task, and because bidirectional attention mechanisms are particularly well-suited to tasks requiring sentence-level understanding.

Rather than relying on a single model, this study deliberately fine-tunes and tests a diverse set of models to better understand how different architectures and language configurations perform on the task. In total, four monolingual BERT models, two multilingual BERT models, and one multilingual RoBERTa model were selected. For English, the monolingual models include distilbert-base-uncased and bert-large-cased. For Italian, bert-base-italian-uncased and bert-base-italian-xxl-cased were used. The multilingual models are bert-base-multilingual-cased and bert-base-multilingual-uncased, while xlm-roberta-base represents the RoBERTa family.

This setup allows for cross-linguistic comparisons regarding whether cased or uncased models perform better on the GED task. It also helps assess how model size influences performance. Naturally, differences in vocabulary size and representational capacity may already bias the results in favor of the larger models.

In addition to investigating architectural factors, this study also aims to evaluate whether multilingual models, which are trained to share representations across languages, can offer performance advantages over monolingual ones. While multilingual models often generalize better for low-resource languages, this benefit comes with a tradeoff: expanding language coverage can dilute vocabulary and parameter capacity, potentially degrading performance on both monolingual and cross-lingual tasks. This effect, referred to as the *curse of multilinguality* (Conneau et al., 2020), is particularly pronounced when model capacity is limited. By applying the same evaluation framework to both monolingual and multilingual variants, this study enables a controlled comparison of their generalization capabilities in the GED setting.

The detailed architecture of the BERT models can be found in Vaswani et al. (2017). In this study, only the aspects relevant to explaining performance shifts between models, and the architectural differences most likely to influence GED performance, will be illustrated and discussed.

Monolingual Models

English As mentioned above, the monolingual models used for English are distilbert-base-uncased [2] and bert-large-cased [3]. As the name suggests, and as the title of Sanh et al. (2019)'s paper makes even clearer, DistilBERT is a distilled version of the original BERT model. It is designed to be faster, cheaper, and lighter while maintaining comparable performance. The model architecture follows the same general structure as BERT but with several simplifications: the number of transformer layers is reduced, and components such as the token-type embeddings and the pooler are removed. These modifications result in significantly faster inference and reduced resource consumption. Despite these reductions, DistilBERT achieves approximately 97% of BERT's original performance according to Sanh et al. (2019), making it an attractive option in scenarios where computational efficiency is a priority.

In contrast, bert-large-cased represents one of the most powerful configurations of the original BERT architecture. It consists of 24 transformer layers (compared to 12 in BERT-base and 6 in DistilBERT), with 1024 hidden units and 16 attention heads per layer for a total of 340 million parameters (Devlin et al., 2019). This significantly larger architecture allows the model to capture more complex linguistic patterns and longer-range dependencies, which can be beneficial in tasks that rely on nuanced grammatical understanding, such as GED. However, this increase in representational power comes at the cost of higher computational requirements. Furthermore, unlike the uncased variant used in DistilBERT, bert-large-cased keeps case distinctions, which may offer advantages in contexts where capitalization conveys meaningful syntactic cues.

Italian For Italian, the selected models include both the base and XXL versions of BERT, specifically trained on Italian text.[4] These models were chosen over alternatives such as AlBERTo because they were pre-trained on corpora comparable to the ones used for distilbert-base-uncased and bert-large-cased, namely Italian Wikipedia and OPUS, providing consistency across the models and the languages used in this study. In contrast, AlBERTo was trained primarily on Italian Twitter data (Polignano et al., 2019), which differs significantly in style, register, and vocabulary, making it less suitable for controlled cross-model comparisons in this context.

As with the English monolingual models, one of the two is smaller in size and differs in whether it preserves information on capitalization. Both Italian models were developed by the DBMDZ team and are based on the original BERT architecture, but they differ significantly in terms of scale and design choices. The smaller model, bert-base-italian-uncased, is a BERT-base variant with 12 layers, 768 hidden units, and approximately 110 million parameters. It does not preserve case information, which simplifies the model but may lead to a loss of syntactic or semantic detail. In

[2] `https://huggingface.co/distilbert/distilbert-base-uncased`

[3] `https://huggingface.co/google-bert/bert-large-cased`

[4] There is no accompanying paper for these models; further information is available on GitHub: `https://huggingface.co/dbmdz/bert-base-italian-cased`.

contrast, bert-italian-xxl-cased is a much larger model, with 48 layers and over 1.3 billion parameters, and it keeps capitalization (DBMDZ, 2020).

Both were pre-trained on a combination of Italian Wikipedia and various OPUS corpora (Tiedemann, 2012), providing exposure to a mix of formal and informal texts. However, bert-italian-xxl-cased's training data was extended with Italian content from the OSCAR corpus (OSCAR Project, 2020).

Multilingual Models

Following the same approach used for the monolingual models, this study also considers both the cased [5] and uncased [6] variants of multilingual BERT (Devlin et al., 2019). For XLM-RoBERTa [7] (Liu et al., 2019), only the base version is included, which preserves capitalization by default. All three models share the same transformer architecture with 12 layers, 768 hidden units, and approximately 110 million parameters, but they differ in their handling of casing, tokenizer design, and pretraining corpora. The mBERT variants were trained on different subsets of Wikipedia: the cased model on the top 104 languages and the uncased one on the top 102, with lowercased text and a correspondingly smaller vocabulary.

XLM-RoBERTa, in contrast, was trained on a significantly larger and more diverse multilingual corpus: the CommonCrawl-based OSCAR dataset (OSCAR Project, 2020), which contains vastly more monolingual data than Wikipedia, particularly benefiting low-resource languages (Conneau et al., 2020). It uses a SentencePiece tokenizer with byte-level encoding, enabling robust handling of rare words and multilingual scripts. While both BERT and RoBERTa share the same transformer structure, RoBERTa introduces key improvements in its pretraining procedure: it removes the next sentence prediction (NSP) objective, uses dynamic masking, and trains on larger corpora with increased batch size and learning rates. These changes contribute to its improved performance.

2.5.2 Generative Model

For the generative stage of this pipeline, this study uses the instruction-tuned variant `qwen2.5-7b-instruct-q4_k_m-00001-of-00002.gguf` of Qwen2.5-7B, a decoder-only transformer developed by Alibaba Cloud (Yang et al., 2024). Qwen2.5[8] models are designed for multilingual instruction-following tasks and long-context generation, with the 7B variant containing approximately 7.6 billion parameters and supporting input lengths up to 128,000 tokens.

In this work, it is used to produce corrected sentences based on the grammatical error classifications generated in the first stage. Thanks to its instruction tuning and multilingual training, Qwen2.5-7B-Instruct is particularly well suited to generate fluent, context-aware corrections in both English and Italian.

Prompt engineering has emerged as a powerful technique in natural language processing, particularly for large language models like Qwen2.5. By carefully crafting instructions, known as prompts, these models can perform a variety of tasks without additional fine-tuning, including classification, translation, summarization, and

[5] `https://huggingface.co/google-bert/bert-base-multilingual-cased`
[6] `https://huggingface.co/google-bert/bert-base-multilingual-uncased`
[7] `https://huggingface.co/docs/transformers/model_doc/xlm-roberta`
[8] `https://huggingface.co/Qwen/Qwen2.5-7B-Instruct-GGUF`

text correction. This capability significantly reduces the reliance on large annotated datasets. A key advantage of prompt engineering is its application in generating artificial (synthetic) data, which is especially beneficial for augmenting training resources in low-resource languages. In the context of this thesis, prompt-based generation is employed to produce grammatically corrected versions of learner sentences, helping to reveal cases where human corrections may have missed alternative possibilities that a discriminative model might classify differently, potentially leading to different and insightful corrections.

Chapter 3

Data and Methodology

This chapter outlines the methodology adopted in this study, detailing the use of the datasets and models introduced in the previous chapter. It begins by describing the preprocessing steps required to construct the aligned datasets. Next, it presents the experimental setup used when participating the MultiGED-2023 shared task, including the hyperparameters used during the fine-tuning of the selected models. Finally, it discusses the prompting strategies employed to generate corrections using the Qwen model.

3.1 Preprocessing and Dataset Alignment

The data provided in the MultiGED-2023 shared task is not sufficient to conduct a detailed investigation into where and why models fail to make correct predictions regarding the grammaticality of a sentence or of individual tokens within it. To enable a more thorough and in-depth analysis, and to explore whether specific grammatical structures underlie the mispredictions, it was necessary to map the MultiGED-2023 development set to their original source datasets. From these, I extracted additional information such as error types and the annotators' corrections. This process allowed me to construct a richer and more comprehensive dataset that not only supports binary evaluations of correctness and incorrectness but also enables a finer-grained analysis of grammatical issues and linguistic patterns. In the following section, I present the preprocessing steps that led to the creation of this final dataset, which I then used to evaluate the performance of the fine-tuned BERT models.

3.1.1 FCE

The FCE dataset contains one XML file per student submission. Each file includes two writing tasks. In contrast with the data found in MultiGED-2023, the FCE files are not tokenized. The data is organized into paragraphs marked by the `<p>` tag. Each paragraph can contain both correct and incorrect forms, with corrections annotated using a nested structures.

The core tags used for error annotation are `<NS>` (error annotation), `<i>` (incorrect form), and `<c>` (correction). Each `<NS>` element contains a `type` attribute describing the error category.

To process the dataset, each file is parsed and all paragraphs are extracted, recursively navigating the XML structure to identify:

- plain text (not within any error tags);
- incorrect tokens (tagged with `<i>`);
- corrections (tagged with `<c>`);
- the error type (from the `type` attribute of `<NS>`).

The preprocessing phase involved standardizing and aligning the FCE and MultiGED datasets to prepare them for token-level error analysis. Tokenization was performed using the `spaCy` library to ensure consistent and linguistically informed segmentation across all texts. Particular attention was given to tokenization inconsistencies, including those involving slashes (e.g., 'Sir/Madam'), punctuation, and contractions, 'haven't' was not split into separate tokens, in the MultiGED 2023s, while other auxiliaries and negations like 'isn't' were tokenized as multiple words. It was, therefore, necessary to normalize and partially adapt SpaCy's tokenization to match the shared task's ones to maximize the alignment across examples.

Sentence segmentation inconsistencies were addressed by merging malformed or short sentences (e.g., list markers or isolated punctuation) with neighboring sentences, and by generating alternative versions to handle potential alignment variability. Similar adjustments were applied to cases involving quotation marks or punctuation at paragraph boundaries. A major source of the segmentation mismatches was the misuse of punctuation in student writing, particularly the use of periods in place of commas. This often led to unintended sentence splits in the MultiGED data that were not present in the FCE source. For example, the sentence 'Finally, I suggest you should try as I did if you have any opportunity. You must like it.' could be perceived as semantically cohesive, yet the period triggers a boundary split that disrupts alignment. To mitigate such issues, both merged and split sentence variants were generated, allowing greater flexibility in matching FCE and MultiGED sentence boundaries.

Alignment between the FCE and MultiGED datasets was performed using exact token sequence matching. In cases where token-level alignment failed, often due to discrepancies in tokenization between the two datasets, a fallback method based on character-level matching was applied. Specifically, space-stripped versions of the sentences were compared character by character to identify matches. This two-step strategy allowed the recovery of additional aligned sentence pairs that would otherwise have been missed, resulting in a more complete and consistently aligned dataset in TSV format.

3.1.2 REALEC

As introduced in Chapter 2, the REALEC dataset contains three files per student submission: a raw text file (`.txt`), a manual annotation file (`.ann`), and a metadata file (`.json`). Similar to the FCE dataset, both the text and annotation files are untokenized and organized into paragraphs.

I started from an available pipeline [1] which began by parsing the `.ann` and `.txt` files. Each annotation file contained manually labeled error spans with corresponding corrections. The raw text was tokenized using the `spaCy` library.

To improve tokenization consistency, special cases such as tokens beginning with hyphens (e.g., `-40`, `-cannot`) were split into separate tokens for the hyphen and the

[1] The original pipeline is available at `https://github.com/Aniezka/REALEC`.

remaining content. To address segmentation mismatches caused by noisy learner data and limitations of automatic sentence splitting, merged sentence variants were created. Each sentence was paired with the next one or two sentences to produce multi-sentence units. This helped mitigate cases where semantically connected sentences were erroneously split due to learner punctuation errors or irregular syntax issues, also observed in the FCE and MultiGED datasets.

Alignment with the MultiGED-2023 dataset followed the same procedure as for the FCE data, starting with exact token sequence matching. When token-level alignment failed, due to differences in tokenization or formatting, a secondary method based on character-level matching was used. Sentences were normalized by removing spaces and escaped characters, and matched using a compressed representation that tracked both token boundaries and label sequences.

Once matched, token indices were reconstructed from compressed sequences, and corresponding labels were attached to the original REALEC tokens. This resulted in a list of fully aligned sentences, where each token entry included the filename, token ID, surface form, up to eight error-correction pairs, and the integrated gold label from the MultiGED dataset.

The final output was a TSV file with one row per token and empty lines separating sentences. Compared to the FCE-aligned TSV format, the REALEC version included additional columns to accommodate multiple annotations per token, reflecting the complexity of the source annotations and the alignment process.

3.1.3 MERLIN

In the MERLIN dataset there is one `.exb` file per student submission. These files include multiple annotation tiers encoding both learner transcriptions and error labels. In particular the following information has been retrieved: `TH1`, `TH2`, `TH1Diff`, `TH2Diff`, and `markable_scheme`. Sentence boundaries were derived from the `sentence` tier, allowing tokens to be grouped accordingly. The resulting data structure mapped each filename to a list of token-level annotations and corresponding sentence segmentations.

During preprocessing, non-linguistic or irrelevant tokens were filtered out. This included tokens marked as unreadable (e.g., `-unreadable-`), visual placeholders (e.g., `-image-`), and tokens that were empty or consisted solely of whitespace.

As for the previously mentioned datasets, to improve alignment robustness, consecutive MERLIN sentences were merged into larger units. For each sentence, up to one or two following sentences were combined, creating a set of single and multi-sentence candidates. For example, the sentence 'Città X, 14.05.2011 Caro Francesco, congratulo con te per avere superato l'esame, ed inoltre con il massimo dei voti!' (*City X, 14.05.2011 Dear Francesco, I congratulate you for passing the exam, and even, with the maximum!*) is stored as a single unit in MultiGED but split into two in MERLIN. Merging helped bridge such inconsistencies.

Initial alignment was attempted using exact string matching between these representations. For unmatched cases, a second alignment strategy was introduced using `difflib.SequenceMatcher`. If a MERLIN sentence and a MultiGED sentence had a similarity score above 0.8 and one was a prefix or suffix of the other, they were treated as a potential match. A third alignment pass was conducted with a relaxed threshold (0.6).

Post-processing involved reconstructing token boundaries from the compressed alignment strings, using token IDs to infer segmentation. Error types and corrections were

then re-integrated by referring back to the initial annotations. The final output was saved in a `.tsv` format, with one row per token. Each row included the following fields: `filename`, `learner`, `TH1`, `TH1Diff`, `TH2`, `TH2Diff`, `markable_scheme`, and `gold_label`.

Finally, malformed sentences were excluded from the final output. These included non-informative or corrupted entries, such as single punctuation marks (e.g., `.`, `!`). The presence of such entries in the MultiGED files suggests that sentence segmentation during dataset construction may have introduced noise. Since these fragments offered no linguistic value for training or evaluation, they were filtered out.

Results will be provided in Section 4.1, including the number of sentences that had a viable match between datasets, as well as the number of error type labels in Section 4.2.4.

3.2 Partecipation in the MultiGED-2023 Shared Task

To participate in the shared task, I followed all standard instructions provided by the organizers. In total, I fine-tuned seven models using the common fine-tuning paradigm: two monolingual models per language, and three multilingual models per language, for a total of six multilingual models. Fine-tuning was performed using the official training splits: for English, the training set of the FCE corpus; for Italian, the training set of the MERLIN corpus. The models were then evaluated on the development and test sets provided by the shared task organizers. Additionally, I evaluated performance on the aligned development set, which contains fewer sentences than the original development set.

Table 3.1 summarizes the amount of training, development, and test data used for each corpus, including sentence, token, and error counts.

Dataset	**Split**	**Sentences**	**Tokens**	**Errors**	**Error Rate**
FCE	Train	28,357	454,736	42,899	9.4%
FCE	Development	2,191	34,748	3,460	10.0%
FCE	Test	2,695	41,932	4,501	10.7%
FCE	Total	33,243	531,416	50,860	9.6%
REALEC	Development	4,067	88,008	8,103	9.2%
REALEC	Test	4,069	89,761	8,505	9.5%
REALEC	Total	8,136	177,769	16,608	9.3%
MERLIN	Train	6,394	80,336	12,190	15.2%
MERLIN	Development	758	9,144	1,211	13.2%
MERLIN	Test	797	10,218	1,492	14.6%
MERLIN	Total	7,949	99,698	14,893	14.9%

Table 3.1: Dataset statistics for training, development, and test splits.

Note: This table is repeated from Section 2.3.2 for convenience.

3.2.1 Fine-Tuning Parameters

All transformer models were fine-tuned on the grammatical error detection task using the Trainer API from the Hugging Face Transformers library. A fixed random seed

(1234) was used across all models to ensure reproducibility and facilitate robust comparisons between them. The learning rate was set to $2 \cdot 10^{-5}$, a commonly used value for fine-tuning pretrained transformer models. This setting balances adaptability to the new task with training stability.

Training was performed for three epochs, which is often sufficient to fine-tune models on smaller or moderately sized datasets without overfitting. Weight decay was set to 0.01 to apply mild regularization and encourage generalization by preventing large parameter updates.

A batch size of 16 was initially chosen for all models. However, this configuration led to training failures for multilingual models and large monolingual models (e.g., bert-base-multilingual-cased, bert-base-italian-xxl-cased). The batch size was progressively reduced until training succeeded, and a batch size of 8 was found to be the minimum value at which fine-tuning could be completed without memory errors. As a result, all large and multilingual models were trained with batch size 8, while smaller monolingual models (e.g., distilbert-base-uncased, bert-base-italian-uncased) were fine-tuned with the original batch size of 16. All other training parameters were kept constant across experiments to ensure comparability.

The multilingual models were fine-tuned separately for English and Italian, rather than jointly, to prevent interference between languages. Given the substantial linguistic differences between English and Italian, cross-lingual transfer was not expected to yield benefits in this task, and language-specific fine-tuning ensured that the model representations remained focused and coherent within each language domain.

3.2.2 Evaluation Metrics

Evaluation starts by presenting the official metrics adopted in the shared task. For each language, the best-performing model is submitted to Codalab, and results are reported in relation to those of other submitted systems.

In addition to the shared task setup, which involves only binary classification, I also include an extended analysis, which maps incorrect tokens to specific error types extracted from the original annotations. In the first stage, model performance is assessed using precision, recall, and $F_{0.5}$.

The Fβ-measure is the harmonic mean of precision and recall. The β parameter differentially weights the importance of recall versus precision. When β >1, recall is given higher weight, while β <1 favors precision.

$$F_{0.5} = (1 + 0.5^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{0.5^2 \cdot \text{Precision} + \text{Recall}}$$

In Educational NLP, following Ng et al. (2014), a β value of 0.5 is commonly preferred. This reflects the pedagogical principle that it is better for a model to confidently identify genuine errors than to flag correct usage as incorrect, which could mislead learners and hinder their progress.

Both micro and macro averages are considered to highlight how imbalanced the classification task is when considering the two labels i and c.

In the second part of the evaluation, the analysis moves beyond binary classification and uses the results of the alignment discussed above to provide information on whether the model correctly identifies specific types of errors in the input. However, this setup introduces a challenge for standard evaluation metrics such as precision and $F_{0.5}$, which rely on both true positives and false positives. In this context, when the

model incorrectly flags a token as erroneous (a false positive), it does so with the simple label *i*. For this reason, there is no gold-standard error type to map it to, since no actual error is present. As a result, it becomes impossible to determine the error category for false positives, making it impossible to compute precision or any F-score.

To address this, a type-level recall-only evaluation is adopted. For each error type defined in the gold annotations, the metric computes the proportion of instances that the model successfully flagged as incorrect. This means that for each type, recall is calculated as the number of correctly predicted erroneous tokens (i.e., the model labeled the token as *i* and it matches a gold *i*) divided by the total number of gold tokens labeled as incorrect for that type.

This approach highlights the model's coverage across different error categories, how well it captures the range of phenomena represented in the dataset. This allows us to assess whether certain error types are systematically underdetected.

3.3 Prompt-Based Correction Generation

3.3.1 Local Language Model Setup

A local version of the Qwen2.5-7B-Instruct model[2] was used to generate corrections for grammatical errors. The model was served through an OpenAI-compatible API and accessed using the OpenAI Python client. A simple wrapper function was written to send prompts and receive responses. The temperature parameter was tested with both 0.6 and 0.3, in order to balance variation in the output with consistency. The final experiments were conducted using a temperature of 0.6, which produced more original but still plausible corrections. The model was queried and run entirely offline, using `qwen2.5-7b-instruct-q4_k_m-00001-of-00002.gguf`, a quantized GGUF version, to reduce resource usage.

As with the discriminative models, all generation was performed on CPU due to the lack of GPU support. This significantly increased generation times, making the process considerably slower compared to standard GPU-accelerated setups.

3.3.2 Prompt Design and Use

Since the primary goal of this thesis is not prompt engineering, a single prompt was selected based on its ability to produce a high number of minimally edited and grammatically plausible corrections. While several variants were tested (see Appendix A), the version below yielded the most consistent and useful results, and is therefore the only one reported in the main text.

The prompt frames the model as a careful **English grammar corrector** or **Italian language editor**, and instructs it to return up to five corrected versions of a given tokenized sentence. Corrections are expected to be minimal, space-separated, and returned on separate lines. If fewer than five corrections are appropriate, empty lines are used to preserve format consistency. The prompt also explicitly instructs the model to avoid explanations or paraphrasing.

To account for occasional failures or empty responses, the model was queried up to `max_attempts=2` per sentence. This retry limit was chosen to balance robustness (recovering from malformed outputs) with computational constraints, as querying large

[2] `https://huggingface.co/Qwen/Qwen2.5-7B-Instruct-GGUF`

language models is both time- and resource-intensive. Increasing the number of retries may improve coverage but would substantially raise processing time and cost.

The final prompt used is shown below:

```
You are a careful Italian language editor.

Given a tokenized sentence, return up to maximum five corrected
    ↪ versions. Each version should:
- Be minimally corrected (fix grammar and lexical errors only)
- Be tokenized (space-separated words)
- Be written on a separate line
- Use deletions if a word is clearly unnecessary or incorrect
- Do not add explanations
- If fewer than five unique corrections are possible, write an
    ↪ empty line for each missing correction

Input: {sentence}
Output:
```

3.3.3 Targeted Generation and Error Type Analysis

For the generation of corrections, only the FCE (English) and MERLIN (Italian) datasets are considered, excluding REALEC. The most meaningful error types are identified based on the results from the best-performing discriminative model for each dataset. If a given error class appears frequently but the model consistently fails to flag it, that class becomes a strong candidate for targeted generation. In such cases, representative sentences containing that error type are extracted and passed to the generative model to assess its capacity to implicitly identify and correct the underlying error.

As shown in the prompt above, the model receives no information about the location or type of the error, ensuring that the correction task remains realistic and unconstrained. Whenever possible, the generated corrections are compared to human corrections from the FCE, REALEC, and MERLIN corpora to assess alignment with gold-standard references.

This comparison also serves a secondary purpose: to investigate whether the predicted labels from the discriminative models, and the corresponding corrections proposed by the generative model, suggest alternative interpretations or valid corrections that may have been overlooked or inconsistently annotated by human annotators. The generated corrections will be further evaluated in Section 5.1.

Both the evaluation results broken down by error type and the error analysis goal, understanding how discriminative and generative models handle ambiguity, inform the selection of error types for correction generation.

3.4 Error Analysis

The final and most substantial part of this work is the error analysis. Since my research questions directly concern how errors are handled, particularly how the discriminative and generative models deal with ambiguity, this analysis is presented as an integral part of the methodology. This section outlines the structure and rationale of the analysis conducted in Chapter 5.

Since the primary focus of this thesis is not fine-tuning or generating corrections, but rather evaluating and interpreting model behavior, the error analysis aims to uncover deeper patterns in error detection. Specifically, it seeks to go beyond simple token-level labeling and examine whether model decisions reflect a more nuanced understanding of linguistic structure. This involves analyzing the distribution of different error types and how they are handled by the models.

The analysis is conducted on the outputs of the best-performing model. Three main scenarios are considered, each reflecting a different type of model misclassification:

- **Sentences containing only false negatives:** These are cases where the model fails to detect all errors present in the sentence. This includes instances where the model detects no errors despite gold labels indicating at least one error, as well as cases where the model makes some correct predictions but misses others (e.g., when a sentence contains three adjacent tokens marked as *i*, but the model correctly identifies only two of them as incorrect). The goal is to assess whether the gold-standard annotation is always justified or whether the sentence, although odd or unconventional, may still be grammatically valid. This type of analysis focuses particularly on high-frequency error types that are often overlooked by the model.

- **Sentences containing both false negatives and false positives:** In these cases, the model detects some incorrect tokens that are not annotated as errors (false positives) and simultaneously misses others that are (false negatives) (e.g., when there are three adjacent tokens labeled as *i*, the model correctly predicts two of the three tokens as *i*, leaving one out and also mistakenly labeling as *i* a token that should be labeled as *c*). The purpose here is to explore whether multiple valid correction strategies exist and whether the model is implicitly offering alternative, plausible revisions. As in the previous case, the error analysis will focus in particular on the false negatives and will group them by error type to better understand which types are being systematically overlooked.

- **Sentences containing only false positives:** These are cases where the model incorrectly flags correct tokens as erroneous. For instance, cases where the model only makes false positive errors, as well as cases where the model makes some correct predictions but also incorrectly labels correct tokens as erroneous (e.g., when the model correctly identifies all three incorrect tokens in a sentence but additionally mislabels a correct token as incorrect). While no specific error type can be mapped to these tokens, the aim is to understand why the model might misidentify them as incorrect. This could be due to internal biases, token-level patterns, or syntactic ambiguity.

The first and second scenarios are analyzed together, highlighting cases of the latter only when relevant or necessary. The model's output in each scenario is then evaluated using three classification categories :

- **Plausible corrections:** Cases where the model's output diverges from the gold label but suggests a reasonable alternative location for the error, pointing to a valid correction path different from the annotated one.

- **Grammatical sentence:** Instances where the model's output is correct because the sentence, or the portion of text corresponding to the specific error type, is in fact grammatical.

- **Model wrong:** Cases in which the model's prediction is incorrect and cannot be traced to any plausible grammatical alternative.

Instead of *Grammatical sentence*, a different classification is used for sentences containing only false positives and correct predictions: **Mislabeled in MultiGED**, which applies to cases where the gold labels clearly contain annotation errors, specifically, instances where there is an actual error but the gold standard incorrectly classifies the token as correct (c).

A further component of the error analysis evaluates the corrections generated by the generative model. Here, the following possible outcomes are identified :

- **Fully Correct**: the model corrected all errors in the sentence, the output aligned with the gold labels, or the original sentence was already correct and required no intervention. This category also includes cases where the model produced a grammatical sentence that differed from the gold correction but could be considered a valid alternative.
- **Partially Correct +**: the output was only partially correct but successfully addressed the specific error type under analysis.
- **Partially Correct -**: the output was partially correct but failed to resolve the target error type.
- **No modification**: the output was identical to the original sentence, even though the sentence could not be considered grammatical.
- **Introducing error**: the model introduced new errors into an otherwise correct or acceptable sentence, most often by removing punctuation entirely.
- **Divergent**: the model altered elements of the sentence in ways that changed the intended meaning or introduced modifications unrelated to error correction, especially when these changes were irrelevant to resolving the targeted error.

For each category, qualitative examples are provided to illustrate typical model behaviour, and an approximate estimate of the generative model's performance is reported.

Chapter 4

Results

This chapter provides a quantitative representation of the models' performance on the task at hand. The chapter is divided into two main parts.

The first part highlights the differences in the original MultiGED-2023 datasets and the datasets, result of the preprocessing, which contains information from the original FCE, RELAEC and MERLIN datasets.

The second part is further subdivided into two sections: the first part focuses on comparing the performance of different models, on the MultiGED datasets, in predicting i and c tokens within sentences; the second section examines in greater detail where the best-performing model for each dataset fail to identify errors, analyzing the recall for each error type.

4.1 Results: Aligning Datasets

As a result of the preprocessing, not all sentences from the MultiGED dataset could be automatically mapped to the original FCE, REALEC, and MERLIN datasets. This is due to systematic differences between the datasets, as well as differing, and at times problematic, preprocessing steps that shaped the current form of the MultiGED data. In the following sections, I will systematically examine these issues, starting with the FCE dataset, followed by REALEC and finally MERLIN. The focus will be on discrepancies in the number of instances between the MultiGED datasets and their original counterparts, with particular attention to cases where alignment could not be achieved.

Table 4.1 presents a preliminary overview of the sentence alignment results, which will be examined in more detail in the following sections.

Dataset	**MultiGED Sentences**	**Automatically Aligned**	**Manually Aligned**	**Total Aligned**	**% Aligned**
FCE	2191	2160	27	2187	99.8%
REALEC	4067	4015	0	4015	98.7%
MERLIN	758	730	0	730	96.3%

Table 4.1: Alignment statistics for the development sets of FCE, REALEC, and MERLIN.

4.1.1 FCE

As shown in Table 4.1, the MultiGED FCE development set contains a relatively high number of sentences, specifically, 2,191. Of these, my alignment process successfully matched 2,160, leaving 31 sentences that could not be automatically aligned.

A number of these unmatched cases were due to inconsistencies in the XML structure of the source files, which complicated sentence extraction. Some files were deeply nested or followed inconsistent annotation patterns, particularly in how corrections and errors were represented. These irregularities made it difficult to programmatically parse the data in a uniform way.

In addition, several mismatches were caused by issues within the MultiGED dataset itself, including typographical errors, tokenization mistakes, and incorrect sentence segmentation. For instance:

```
FCE split:      I|have|asked|to|write|a|composition|regarding|the|
    ↪ question|it|is|believed|that|shopping|is|not|always|enjoyable|.
MultiGED split: I|have|asked|to|write|a|composition|regarding|the|
    ↪ questionit|is|believed|that|shopping|is|not|always|enjoyable|.
```

As shown, the MultiGED version incorrectly merges the words ‘question’ and ‘it’ into a single token (‘questionit’), preventing a direct alignment. In cases like this, when the sentence could be reliably traced back to the original FCE source, I manually inserted the corresponding sentence and token-level annotations into the final TSV file. However, not all problematic sentences could be manually added due to irregularities in the FCE structure or limitations in my code’s handling of edge cases. This affected only three sentences.

Additionally, I deliberately excluded one MultiGED sentence from the final set. This sentence ‘!’, labeled as correct, originated from a flawed sentence split in the MultiGED dataset. The original FCE file contains ‘Yeah!!!’ (with three exclamation marks), but the MultiGED version splits it into two separate sentences: ‘Yeah!!’ and ‘!’. The latter, of course, does not correspond to any meaningful sentence in the source data and appears to be caused by erroneous preprocessing.

After preprocessing, the aligned dataset consists of 2,187 sentences.

4.1.2 REALEC

The REALEC portion of the MultiGED dataset contains 4,067 sentences, as noticeable in Table 4.1. After alignment, however, only 4,015 sentences could be successfully matched, leaving 52 sentences that could not be aligned.

The majority of these alignment failures, 42 sentences, are due to incorrect preprocessing in the MultiGED dataset itself, where several sentences contain duplicated tokens. For example:

```
[‘People’, ‘People’, ‘who’, ‘who’, ‘create’, ‘create’,
‘illegal’, ‘illegal’, ‘pirate’, ‘pirate’,
 ‘copies’, ‘copies’, ‘are’, ‘are’, ‘responsible’,
‘responsible’, ‘for’, ‘for’, ‘this’, ‘this’,
 ‘and’, ‘and’, ‘should’, ‘should’, ‘be’, ‘be’, ‘punished’,
 ‘punished’, ‘.’, ‘.’]
```

An additional 10 sentences were excluded due to tokenization inconsistencies, such as missegmented special characters. These issues led to misalignment between the source and the MultiGED version, making accurate matching impossible. For instance, the abbreviation ‘U.S.’ is incorrectly split into separate tokens:

```
['The', 'following', 'bar', 'chart', 'illustrates',
'the', 'use', 'of', 'major', 'social', 'networks',
':', 'Facebook', ',', 'Instagram', 'and',
'LinkedIn', 'among', 'U.S', '.', 'adults', 'divided', 'into', '4',
'age', 'groups', '.']
```

These errors are not trivial to fix automatically and introduce inconsistencies that prevent reliable alignment. As a result, the affected sentences were excluded from the final dataset. However, given the relatively small number of affected sentences, their removal can be seen as a form of noise reduction rather than a significant loss of usable data.

4.1.3 MERLIN

The original MERLIN dataset used for MultiGED contained 758 sentences. However, as one could see in Table 4.1, after preprocessing, 730 sentences were successfully aligned.

As with previous cases, most instances of misalignment stem from inconsistencies within the MultiGED dataset itself. In several cases, I deliberately excluded specific sentences that introduced noise into the data. These included, for example, four single-token sentences such as '!', as well as entries that did not correspond to any real sentence in the original dataset but were instead artifacts of preprocessing, often created by the erroneous merging of different parts of separate sentences within the same text.

I also excluded sentences that had been mistakenly cut mid-sentence, as they would have contributed fragmented and uninformative data. These decisions were made to preserve the overall quality and reliability of the aligned dataset.

4.2 Shared Task Results

4.2.1 Results: MultiGED-2023 Datasets

The following sections reports the results on the original MultiGED datasets, excluding those obtained from the aligned versions.

English(FCE)

As shown in Table 4.2, the models exhibit generally consistent and comparable performance across both precision and recall for the two target labels. The table is organized with models as rows and evaluation metrics—precision (P), recall (R), and $F_{0.5}$—grouped by class label (c, correct, and i, incorrect) in the columns. All results are reported at the token level.

For the majority class c, all models perform strongly, with precision ranging from 0.93 to 0.94 and recall from 0.97 to 0.98. In contrast, the minority class i shows more significant variability and limitations in model performance. Among the models, XLM-RoBERTa-base achieves the highest precision at 0.75, followed by BERT-base-multilingual-uncased (0.72), while all other models reach 0.71. Recall values for class i are drastically lower, ranging from 0.40 to 0.49, with BERT-large-cased attaining the highest recall (0.49), followed by XLM-RoBERTa-base (0.43).

Nevertheless, the performance gap is marginal, especially because only one seed was used, and all models achieve comparable results. The $F_{0.5}$ score for the c label

remains consistently high at 0.94 across most models, with a slight edge for BERT-large-cased (0.95). For the *i* label, which represents the more challenging minority class, scores range from 0.61 (lowest, by DistilBERT-base-uncased) to 0.65 (highest, shared by BERT-large-cased and XLM-RoBERTa-base).

Despite identical $F_{0.5}$ scores for class *i* in XLM-RoBERTa-base and BERT-large-cased, I consider XLM-RoBERTa-base, the model highlighted in green in the table, the most effective model in this context. Given that educational NLP tasks often prioritize precision, so as to minimize false positive corrections that may mislead learners, the superior precision of XLM-RoBERTa-base in the minority class makes it a more reliable choice, even at the expense of slightly lower recall.

Model	**Label c**				**Label i**			
	P	R	$F_{0.5}$	supp.	P	R	$F_{0.5}$	supp.
distilbert-base-uncased	**0.93**	**0.98**	**0.94**	**31288**	**0.71**	**0.40**	**0.61**	**3460**
bert-large-cased	0.94	0.97	0.95	31288	0.71	0.49	0.65	3460
bert-base-multilingual-cased	0.93	0.98	0.94	31288	0.71	0.40	0.62	3460
bert-base-multilingual-uncased	0.93	0.98	0.94	31288	0.72	0.42	0.63	3460
xlm-roberta-base	**0.94**	**0.98**	**0.94**	**31288**	**0.75**	**0.43**	**0.65**	**3460**

Table 4.2: Precision, Recall, $F_{0.5}$ and support for Label *c* and Label *i* for MultiGED (FCE)

These trends are reflected in the average scores, displayed in Table 4.3, where BERT-large-cased and XLM-RoBERTa-base emerge as the best-performing models. Even in this case, XLM-RoBERTa-base is considered the most effective overall, for the same reason discussed above: its superior precision in the minority class makes it particularly suitable for educational NLP applications. In contrast, the lowest-performing model is DistilBERT-base-uncased, as highlighted in bold red in the table.

Model	**Micro avg**			**Macro avg**			**Support**
	P	R	$F_{0.5}$	P	R	$F_{0.5}$	
distilbert-base-uncased	**0.92**	**0.92**	**0.92**	**0.82**	**0.69**	**0.78**	**34748**
bert-large-cased	0.93	0.93	0.93	0.83	0.73	0.80	34748
bert-base-multilingual-cased	0.92	0.92	0.92	0.82	0.69	0.78	34748
bert-base-multilingual-uncased	0.92	0.92	0.92	0.83	0.70	0.79	34748
xlm-roberta-base	**0.92**	**0.92**	**0.92**	**0.84**	**0.70**	**0.80**	**34748**

Table 4.3: Micro and Macro Averages of Precision, Recall, and $F_{0.5}$ for MultiGED (FCE)

To gain a more detailed understanding of model performance on the MultiGED-FCE dataset, Figure 4.1 presents the confusion matrices for the five evaluated models. The matrices are organized in rows: the first row includes DistilBERT and BERT-large-cased, the second row shows mBERT-cased and mBERT-uncased, and the final row presents XLM-RoBERTa. In each table, the rows correspond to the true (gold) labels, *c* (correct) and *i* (incorrect), while the columns indicate the model predictions.

To help visual interpretation, the diagonal cells, representing true positives, are shaded in light blue. These cells indicate the number of correctly classified instances for each label. High values along the diagonal signal strong classification performance, while off-diagonal values represent misclassifications.

As previously noted, all models demonstrate consistently strong performance on the majority class c, with true positives far outnumbering false negatives. For example, even one of the lowest-performing models overall, DistilBERT-base-uncased, correctly classifies 30,727 instances of class c, while the best-performing model by macro-averaged $F_{0.5}$, BERT-large-cased, correctly identifies 30,616 such instances.

In contrast, classification of the minority class i exhibits greater variation across models. BERT-large-cased and XLM-RoBERTa-base both achieve relatively high numbers of true positives, 1,703 and 1,497 respectively, indicating stronger performance on identifying actual errors. However, these two models differ in their trade-off between recall and precision. BERT-large-cased, having the highest recall, correctly identifies more true positives, but also produces more false positives (672). XLM-RoBERTa-base, on the other hand, has fewer false positives (498), leading to higher precision but slightly lower recall.

Interestingly, mBERT-cased performs slightly worse than its uncased counterpart, despite the expectation that cased models would benefit from capitalization cues, which are often informative for grammatical error detection. mBERT-cased and DistilBERT-base-uncased achieve lower true positive counts for class i (1,401 and 1,388 respectively), with correspondingly higher false negatives.

	Pred c	**Pred i**
True c	30727	561
True i	2072	1388

(a) DistilBERT-base-uncased

	Pred c	**Pred i**
True c	30616	672
True i	1757	1703

(b) BERT-large-cased

	Pred c	**Pred i**
True c	30731	557
True i	2059	1401

(c) m-BERT-cased

	Pred c	**Pred i**
True c	30746	542
True i	2005	1455

(d) m-BERT-uncased

	Pred c	**Pred i**
True c	30790	498
True i	1963	1497

(e) XLM-RoBERTa

Figure 4.1: Confusion matrices for five models on MultiGED-FCE.

English (REALEC)

In the REALEC dataset, as shown in Table 4.4, the models' behavior in predicting the correct tokens within a sentence remains broadly consistent with what was observed on the FCE dataset. However, performance on the minority class i deteriorates significantly. Precision does not exceed 0.48 for any model; this peak is reached by XLM-RoBERTa, which had previously achieved a substantially higher precision of 0.75 on the FCE dataset. Recall is similarly low, with the highest value being 0.40, obtained by BERT-large-cased. The remaining models yield a recall of 0.34, except for XLM-RoBERTa, which reaches 0.36. The $F_{0.5}$ score for the i class remains flat at 0.43 across

all models, with the exception of XLM-RoBERTa.

In contrast, BERT-large-cased, which was the second best model for FCE, underperforms across most metrics, resulting in the lowest macro and micro $F_{0.5}$ scores overall, as shown in Table 4.5. Even in this more challenging setting, XLM-RoBERTa maintains the highest precision among all models, which directly translates into the highest $F_{0.5}$ score for the minority class. This advantage is also reflected in the macro-average, making XLM-RoBERTa the best-performing model on the REALEC.

Model	**Label c**				**Label i**			
	P	R	$F_{0.5}$	supp.	P	R	$F_{0.5}$	supp.
distilbert-base-uncased	0.93	0.95	0.94	79905	0.46	0.34	0.43	8103
bert-large-cased	**0.94**	**0.94**	**0.94**	**79905**	**0.44**	**0.40**	**0.43**	**8103**
bert-base-multilingual-cased	0.93	0.95	0.94	79905	0.46	0.34	0.43	8103
bert-base-multilingual-uncased	0.93	0.96	0.94	79905	0.47	0.34	0.43	8103
xlm-roberta-base	**0.93**	**0.96**	**0.94**	**79905**	**0.48**	**0.36**	**0.45**	**8103**

Table 4.4: Precision, Recall, $F_{0.5}$ and support for Label *c* and Label *i* for MultiGED (REALEC)

Model	**Micro avg**			**Macro avg**			**Support**
	P	R	$F_{0.5}$	P	R	$F_{0.5}$	
distilbert-base-uncased	0.90	0.90	0.90	0.70	0.65	0.68	88008
bert-large-cased	**0.89**	**0.89**	**0.89**	**0.69**	**0.67**	**0.68**	**88008**
bert-base-multilingual-cased	0.90	0.90	0.90	0.70	0.65	0.68	88008
bert-base-multilingual-uncased	0.90	0.90	0.90	0.70	0.65	0.68	88008
xlm-roberta-base	**0.90**	**0.90**	**0.90**	**0.70**	**0.66**	**0.69**	**88008**

Table 4.5: Micro and Macro Averages of Precision, Recall, and $F_{0.5}$ for MultiGED (REALEC)

As shown in Figure 4.2, as for the FCE dataset, all models correctly classify the majority class *c* ; no model has, indeed, less than 75,000 true positives when classifying the majority class. Among them, m-BERT-uncased has the highest number of correct predictions for class *c* (76,820). However, even though its performance outbests the other models on the majority class, its performance on the minority class *i* is disappointing.

In contrast, BERT-large-cased achieves the highest number of true positives for class *i* (3,265), followed by XLM-RoBERTa (2,919). This suggests a comparatively better ability to identify minority-class instances. However, when considering the total number of tokens that should have been labeled as *i*, the overall performance remains limited.

It is important to note that the models' performances on the REALEC dataset were expected to be worse than those on the FCE. All models were trained exclusively on the FCE portion of the MultiGED dataset, with REALEC used solely as an out-of-domain test set. As a result, the domain mismatch, combined with variations in text genre, error types, and linguistic patterns, likely contributed to the drop in performance, particularly on the minority class *i*.

	Pred c	Pred i
True c	76708	3197
True i	5293	2810

(a) DistilBERT-base-uncased

	Pred c	Pred i
True c	75830	4075
True i	4838	3265

(b) BERT-large-cased

	Pred c	Pred i
True c	76701	3204
True i	5301	2802

(c) m-BERT-cased

	Pred c	Pred i
True c	76820	3085
True i	5341	2762

(d) m-BERT-uncased

	Pred c	Pred i
True c	76768	3137
True i	5184	2919

(e) XLM-RoBERTa

Figure 4.2: Confusion matrices for five models on MultiGED-REALEC.

Italian (MERLIN)

Finally, in relation to the Italian dataset, as shown in Table 4.6 and 4.7, all models achieve consistently high performance on the majority class label *c*, with precision and recall values close to or exceeding 0.90, similar to the English models. The highest $F_{0.5}$ score for label *c* is achieved by bert-base-italian-xxl-cased (0.95), though all models demonstrate strong performance on this class, with $F_{0.5}$ scores ranging from 0.93 to 0.95.

However, performance on the minority class label *i* varies considerably between models. While bert-base-italian-xxl-cased again leads with an $F_{0.5}$ score of 0.80, other models like bert-base-multilingual-uncased perform substantially worse ($F_{0.5} = 0.59$), mainly due to a lower recall (0.35). xlm-roberta-base shows a relatively strong balance, with a precision of 0.83 and recall of 0.52 on label *i*, resulting in an $F_{0.5}$ score of 0.74. These results suggest that larger and more specialized models are better able to generalize to underrepresented classes, while smaller or uncased variants struggle more to capture the patterns. Interestingly, the Italian dataset shows greater variability in performance between monolingual and multilingual models compared to FCE, where such differences were minimal. This may point to a sensitivity in multilingual models when applied to lower-resource languages like Italian, potentially due to differences in pretraining exposure or even initialization seeds.

Model	Label c				Label i			
	P	R	$F_{0.5}$	supp.	P	R	$F_{0.5}$	supp.
bert-base-italian-uncased	0.92	0.98	0.93	7933	0.78	0.44	0.67	1211
bert-base-italian-xxl-cased	**0.94**	**0.98**	**0.95**	**7933**	**0.86**	**0.63**	**0.80**	**1211**
bert-base-multilingual-cased	0.92	0.97	0.93	7933	0.78	0.47	0.69	1211
bert-base-multilingual-uncased	**0.90**	**0.97**	**0.92**	**7933**	**0.72**	**0.35**	**0.59**	**1211**
xlm-roberta-base	0.93	0.98	0.94	7933	0.83	0.52	0.74	1211

Table 4.6: Precision, Recall, $F_{0.5}$ and support for Label *c* and Label *i* for MultiGED (MERLIN)

Model	Micro avg			Macro avg			Support
	P	R	$F_{0.5}$	P	R	$F_{0.5}$	
bert-base-italian-uncased	0.91	0.91	0.91	0.85	0.71	0.80	9144
bert-base-italian-xxl-cased	**0.93**	**0.93**	**0.93**	**0.90**	**0.80**	**0.88**	**9144**
bert-base-multilingual-cased	0.91	0.91	0.91	0.85	0.72	0.81	9144
bert-base-multilingual-uncased	**0.89**	**0.89**	**0.89**	**0.81**	**0.66**	**0.75**	**9144**
xlm-roberta-base	0.92	0.92	0.92	0.88	0.75	0.84	9144

Table 4.7: Micro and Macro Averages of Precision, Recall, and $F_{0.5}$ for MultiGED (MERLIN)

Figure 4.3 illustrates how each model distinguishes between correct and incorrect tokens in the MultiGED-MERLIN dataset. Across all five systems, the majority class *c* is identified with high accuracy, with true positive counts consistently above 7,750. Among them, BERT-XXL-Italian-cased stands out for achieving the highest correct classifications for both classes, especially for the minority label *i*, where it correctly predicts 768 instances—substantially more than any other model.

On the other end of the spectrum, m-BERT-uncased misclassifies the largest number of minority-class tokens, correctly predicting only 429 out of over 1,200. This contrasts with XLM-RoBERTa, which captures 637 true positives for class *i* and keeps false negatives relatively lower. While the multilingual models show acceptable performance on class *c*, their ability to detect label *i* lags behind the Italian-specific counterparts.

As with the figures for FCE and REALEC, the layout here follows the same structure: the monolingual models are shown first, followed by the multilingual BERT models, and finally XLM-RoBERTa.

	Pred c	Pred i
True c	7783	150
True i	671	540

(a) BERT-base-Italian-uncased

	Pred c	Pred i
True c	7816	117
True i	443	768

(b) BERT-XXL-Italian-cased

	Pred c	Pred i
True c	7770	163
True i	630	581

(c) m-BERT-cased

	Pred c	Pred i
True c	7767	166
True i	782	429

(d) m-BERT-uncased

	Pred c	Pred i
True c	7809	124
True i	574	637

(e) XLM-RoBERTa

Figure 4.3: Confusion matrices for five models on MultiGED-MERLIN.

4.2.2 Results: Codalab

The results presented in Figure 4.4 correspond to the official test set submissions uploaded to CodaLab as part of active participation in the MultiGED 2023 shared task. These scores were produced by the shared task organizers and represent the outcome of the final evaluation on the held-out test sets. The reported values refer specifically to performance on the i label, evaluated using precision, recall, and $F_{0.5}$ score.

I decided to submit the results obtained by the XLM-RoBERTa architecture. Although this system was not necessarily the top performer in every configuration, it was selected in alignment with the task guidelines, which emphasized or encouraged the use of multilingual models. Accordingly, XLM-RoBERTa-base was used for all submissions made.

The approach achieved strong results across all tracks, ranking second in each subtask. In the English FCE track, it obtained the highest recall (0.43) among all participants and an $F_{0.5}$ score of 0.6472. On the English REALEC task, it again reached the top recall (0.3523) and ranked second in precision, yielding a competitive $F_{0.5}$ score of 0.4504. For the Italian dataset, the system demonstrated balanced performance, with a precision of 0.8403 and recall of 0.5818, resulting in an $F_{0.5}$ of 0.7717.

While the results are broadly comparable to those obtained on the development set, particularly on the i label, they are consistently lower across all sub-tasks. This decline is expected, as the official evaluation was conducted on a fully unseen test set with potentially higher linguistic variability, which could have negatively impacted generalization in the task.

Results English FCE

#	User	Entries	Date of Last Entry	Team Name	Precision ▲	Recall ▲	F05 Score ▲
1	suttergustavo	39	08/08/23		0.831 (1)	0.40 (2)	0.6824 (1)
2	elisabetta.dentico	10	06/28/25	EliThesis	0.743 (2)	0.43 (1)	0.6472 (2)
3	larsbungum	59	04/02/23	NTNU-TRH	0.613 (3)	0.37 (3)	0.5400 (3)
4	RyszardStaruch	36	12/16/24	AMU-CAI	0.396 (4)	0.28 (4)	0.3672 (4)
5	adnanlabib15	192	07/17/23		- (5)	- (5)	- (5)
6	glopezlatouche	45	06/13/24		- (5)	- (5)	- (5)

Results English REALEC

#	User	Entries	Date of Last Entry	Team Name	Precision ▲	Recall ▲	F05 Score ▲
1	suttergustavo	39	08/08/23		0.5692 (1)	0.3480 (2)	0.5050 (1)
2	elisabetta.dentico	10	06/28/25	EliThesis	0.4841 (2)	0.3523 (1)	0.4504 (2)
3	larsbungum	59	04/02/23	NTNU-TRH	0.3894 (3)	0.3035 (3)	0.3685 (3)
4	RyszardStaruch	36	12/16/24	AMU-CAI	0.3782 (4)	0.1999 (4)	0.3209 (4)
5	adnanlabib15	192	07/17/23		- (5)	- (5)	- (5)
6	glopezlatouche	45	06/13/24		- (5)	- (5)	- (5)

Results Italian

#	User	Entries	Date of Last Entry	Team Name	Precision ▲	Recall ▲	F05 Score ▲
1	suttergustavo	39	08/08/23		0.9076 (1)	0.5858 (1)	0.8177 (1)
2	elisabetta.dentico	10	06/28/25	EliThesis	0.8403 (2)	0.5818 (2)	0.7717 (2)
3	glopezlatouche	45	06/13/24		0.7350 (3)	0.4444 (3)	0.6500 (3)
4	RyszardStaruch	36	12/16/24	AMU-CAI	0.3867 (4)	0.1991 (4)	0.3254 (4)
5	larsbungum	59	04/02/23	NTNU-TRH	0.1549 (5)	0.0972 (5)	0.1385 (5)
6	adnanlabib15	192	07/17/23		- (6)	- (6)	- (6)

Figure 4.4: Results from CodaLab for all Test Sets

4.2.3 Results: Pre-processed Datasets

The sections below summarize the key results on the aligned datasets; full details are available in Appendix B.

English (FCE)

Table 4.8 presents the micro and macro average performance scores of the models on the aligned FCE dataset. As expected, the results suggest that the removal of certain cases did not substantially affect the overall evaluation outcomes. A more detailed analysis, including per-label results and confusion matrices, is provided in Appendix B.1.

Model	Micro avg			Macro avg			Support
	P	R	$F_{0.5}$	P	R	$F_{0.5}$	
distilbert-base-uncased	**0.92**	**0.92**	**0.92**	**0.82**	**0.69**	**0.78**	**34703**
bert-large-cased	0.93	0.92	0.92	0.83	0.73	0.80	34704
bert-base-multilingual-cased	0.92	0.92	0.92	0.82	0.69	0.78	34704
bert-base-multilingual-uncased	0.92	0.92	0.92	0.83	0.70	0.79	34704
xlm-roberta-base	**0.92**	**0.92**	**0.92**	**0.84**	**0.70**	**0.80**	**34703**

Table 4.8: Processed-FCE: Micro and macro averages for Precision, Recall, and $F_{0.5}$.

English (REALEC)

Macro averages, as shown in Table 4.9, remained unchanged for all models (0.69 for most, 0.68 in the original), reinforcing the conclusion that performance on the minority class i remains challenging, and that the modifications introduced by preprocessing did not meaningfully improve model sensitivity. More detailed information can be found in Appendix B.2.

Model	Micro avg			Macro avg			Support
	P	R	$F_{0.5}$	P	R	$F_{0.5}$	
distilbert-base-uncased	0.90	0.90	0.90	0.70	0.65	0.69	86362
bert-large-cased	**0.89**	**0.89**	**0.89**	**0.69**	**0.67**	**0.69**	**86362**
bert-base-multilingual-cased	0.90	0.90	0.90	0.70	0.65	0.69	86362
bert-base-multilingual-uncased	0.90	0.90	0.90	0.70	0.65	0.69	86362
xlm-roberta-base	**0.90**	**0.90**	**0.90**	**0.71**	**0.66**	**0.69**	**86362**

Table 4.9: Processed-REALEC: Micro and macro averages for Precision, Recall, and $F_{0.5}$.

Italian (MERLIN)

The macro and micro averages (4.10) remain stable, confirming that the filtered dataset preserves the original label distribution and does not distort model evaluation. These findings suggest that the preprocessed dataset is a reliable substitute for the full set, particularly when the goal of the study is to analyze error types and to have a thorough evaluation of the models' failures above the binary classification surface. More details in Appendix B.3.

Model	Micro avg			Macro avg			Support
	P	R	$F_{0.5}$	P	R	$F_{0.5}$	
bert-base-italian-uncased	0.91	0.91	0.91	0.84	0.71	0.80	8850
bert-base-italian-xxl-cased	**0.93**	**0.93**	**0.93**	**0.90**	**0.81**	**0.88**	**8850**
bert-base-multilingual-cased	0.91	0.91	0.91	0.85	0.73	0.81	8850
bert-base-multilingual-uncased	**0.89**	**0.89**	**0.89**	**0.81**	**0.66**	**0.76**	**8850**
xlm-roberta-base	0.92	0.92	0.92	0.88	0.75	0.84	8850

Table 4.10: Processed-MERLIN: Micro and macro averages split into Precision, Recall, and $F_{0.5}$.

4.2.4 Results: Error Type

The following section presents a detailed analysis of the results obtained in the binary classification task of the MultiGED 2023 benchmark. Specifically, it examines how the prediction errors made by the best-performing model for each dataset are distributed across the various error categories defined in the original FCE, REALEC, and MERLIN corpora. This analysis aims to identify which types of grammatical or lexical errors pose greater challenges to the model, and whether consistent patterns of misclassification emerge across datasets.

While earlier sections compared multiple models on the development sets, the following error-type analyses focus exclusively on XLM-RoBERTa. This model was the model used for the official participation in the MultiGED 2023 shared task. This allows for a more interpretable examination of error distributions without the additional complexity of cross-model variation.

The tables in each section report recall scores for XLM-RoBERTa across various grammatical error types within each dataset. Each row corresponds to a specific error type, with performance metrics presented per category. The tables are structured as follows:

- **Error Type**: the grammatical or lexical category, expressed in plain English rather than abbreviated codes (e.g., *Verb Agreement* instead of AGV).
- **Gold**: the number of gold-standard instances for that error type.
- **TP**: the number of true positives identified by XLM-RoBERTa.
- **FN**: the number of false negatives, instances where the model failed to identify an error.
- **Recall**: the recall score for that error type, calculated as TP / (TP + FN).

A special row labelled *UNKNOWN* appears in each table. This category includes cases where the MultiGED 2023 dataset annotates a token as erroneous (labelled i), even though no corresponding error type is defined in the original FCE, REALEC, or MERLIN datasets. These instances likely stem from inconsistencies between the MultiGED annotations and the source corpora. As a result, the model's recall on this category is particularly low: 0.38 for FCE, 0.13 for REALEC, and 0.55 for MERLIN. This category may include both spurious error annotations (i.e., cases incorrectly labelled as errors) and tokens that could plausibly be mappped to existing error types but are instead left uncategorised.

The final row in each table aggregates performance across all error types, showing the total number of annotated instances and the overall recall score for the model.

Analysis of Model Recall Performance on FCE

As shown in Table 4.11, XLM-RoBERTa achieves an overall recall of 0.43 on the FCE dataset. This suggests moderate success in identifying learner errors but also indicates considerable room for improvement, particularly in more complex or less superficial categories.

The most common category, *Spelling*, includes 331 instances. The model performs very well on this type of error with a recall of 0.92, indicating strong coverage of surface-level orthographic errors. In contrast, errors due to *Confusion Spelling*, where the error

involves a real but contextually inappropriate word, still cause challenges, with a recall of only 0.62, substantially lower than the more basic spelling mistakes.

The model also struggles with high-frequency but linguistically complex categories. For instance, *Verb Tense* (235 instances) shows a recall of just 0.31, reflecting the challenge of temporal and discourse-level reasoning. Similarly, *Word Order*, another common error type with 327 instances, shows a recall of only 0.30. A dedicated section later in this chapter further investigates potential reasons for these low scores.

Several moderately frequent error types, including *Replacing Pronoun*, *Replacing Determiner*, and *Replacing Conjunction*, perform even worse, with recall scores ranging from 0.12 to 0.17. The general *Replacing* category, which includes 150 instances, also shows very poor performance, with a recall of just 0.17. This label is not associated with any specific part of speech and typically requires the model to perform high-level semantic rewriting, which proves difficult given limited contextual cues.

Similarly, *Argument Structure* errors (91 instances) has one of the lowest recall scores (0.10), reflecting difficulties in modelling grammatical relations and verb valency beyond surface token interactions.

Verb Agreement and *Noun Agreement*, with recall scores of 0.75 and 0.48 respectively, are potentially interesting because they often allow for multiple valid corrections. In such cases, the model might produce a valid correction that differs from the one provided by the annotators. As a result, its true performance on these error types may be better than what the recall score suggests. Further qualitative analysis is needed to assess whether these alternative corrections are indeed acceptable.

Taken together, the results show that XLM-RoBERTa handles frequent, surface-level errors like spelling and basic inflection well, but struggles with more complex or context-sensitive categories such as verb tense, word order, and argument structure. Errors involving functional words, like determiners, punctuation, and conjunctions, also yield surprisingly low recall, despite their high frequency. Moreover, several rare categories show zero recall, likely due to data sparsity.

These findings are consistent with the idea that models may be struggling with abstract or ambiguous learner errors. In some cases, such as noun or verb agreement, the model may generate valid corrections that differ from the annotated ones, suggesting its performance may be underestimated. Further qualitative analysis and targeted augmentation is needed to improve generalisation on these harder cases.

Error Type	**Gold**	**TP**	**FN**	**Recall**
Missing Verb	1	1	0	1.00
Incorrect Quantifier	2	2	0	1.00
Noun Countability	9	9	0	1.00
Incorrect Verb Inflection	30	30	0	1.00
Incorrect Anaphoric	2	2	0	1.00
Derived Pronoun	15	14	1	0.93
Spelling	331	304	27	0.92
Incorrect Form Noun Plur.	11	10	1	0.91
Derived Verb	7	6	1	0.86
Determiner Agreement	7	6	1	0.86
Derived Determiner	4	3	1	0.75
Form Adverb	4	3	1	0.75
Missing Determiner	8	6	2	0.75
Verb Agreement	**55**	**41**	**14**	**0.75**

Continued on next page

Error Type	**Gold**	**TP**	**FN**	**Recall**
Adjective Form	3	2	1	0.67
Derived Adjective	46	30	16	0.65
Derived Adverb	23	15	8	0.65
Derived Noun	34	23	11	0.68
Unnecessary Determiner	8	5	3	0.62
Confusion Spelling	**50**	**31**	**19**	**0.62**
Missing Pronoun	5	3	2	0.60
Replacing Punctuation	241	141	100	0.59
Replacing Preposition	205	106	99	0.52
American Spelling	21	11	10	0.52
Verb Form	106	54	52	0.51
Missing	4	2	2	0.50
Noun Form	57	28	29	0.49
Noun Agreement	**69**	**33**	**36**	**0.48**
UNKNOWN	559	214	345	0.38
Inappropriate Register	8	3	5	0.38
Verb Tense	**235**	**73**	**162**	**0.31**
Word Order	**327**	**99**	**228**	**0.30**
Idiom	53	15	38	0.28
Unnecessary Punctuation	24	6	18	0.25
Replacing Adjective	48	12	36	0.25
Unnecessary Verb	4	1	3	0.25
Missing Punctuation	28	7	21	0.25
Replacing Noun	101	24	77	0.24
Replacing Verb	**205**	**46**	**159**	**0.22**
Incorrect Negation	23	5	18	0.22
Pronoun Agreement	5	1	4	0.20
Replacing	150	25	125	0.17
Replacing Adverb	59	10	49	0.17
Wrong Quantifier	6	1	5	0.17
Replacing Quantifier	13	2	11	0.15
Replacing Pronoun	55	7	48	0.13
Replacing Determiner	41	5	36	0.12
Argument Structure	91	9	82	0.10
Quantifier Agreement	1	0	1	0.00
Replacing Conjunction	19	0	19	0.00
Unnecessary/Redundant	5	0	5	0.00
Derived Preposition	1	0	1	0.00
Unnecessary Pronoun	6	0	6	0.00
Unnecessary Conjunction	3	0	3	0.00
Missing Preposition	1	0	1	0.00
Determiner Form	6	0	6	0.00
Unnecessary Adverb	3	0	3	0.00
Unnecessary Preposition	2	0	2	0.00
TOTAL	**3449**	**1489**	**1960**	**0.43**

Table 4.11: Recall per error type for XLM-RoBERTa on the FCE dataset (sorted by recall)

Analysis of Model Recall Performance on REALEC

XLM-RoBERTa achieves an overall recall of 0.36 on the REALEC dataset. While the model handles rare error types such as *Adj_as_collective*, *Adverbs*, and *Vocabulary* with perfect recall, these categories include only a single instance each and thus offer limited diagnostic value.

Similar to the FCE dataset, *Spelling* is the most frequent category in REALEC (1503 instances), and the model achieves a high recall of 0.84, comparable to its 0.92 recall on the same category in FCE. This suggests that the model is consistently effective at surface-level orthographic errors across datasets.

However, its performance drops notably on other frequent categories. For instance, *Articles* (851 instances) shows a recall of 0.45, consistent with the model's struggles with context-dependent grammatical choices in FCE as well. *Word_order* (507 instances) yields a particularly low recall of 0.11, which is even lower than the 0.30 recall observed in FCE.

Punctuation is another problematic class: despite its surface-level nature and high frequency (644 instances), the model achieves only 0.04 recall. This underperformance could derive from annotation inconsistencies, tokenization issues, or low sensitivity to punctuation tokens during encoding.

Grammatical categories such as *Tense_choice* (368 instances, 0.18 recall) illustrate the challenge of correcting temporal errors when contextual information is limited. Since the datasets are segmented into isolated sentences, the model lacks access to wider discourse cues that are often necessary to determine the appropriate verb tense, potentially explaining the low performance in this category, which mirrors similar patterns observed in the FCE results.

In terms of grammatical agreement, results are diverse. *Noun_number* errors, which test the model's ability to detect plural-singular mismatches, yield a recall of 0.38, suggesting modest success but also frequent oversight. *Agreement_errors* errors show slightly better performance, with a recall of 0.67, though this still indicates substantial room for improvement. Notably, as mentioned before, these cases deserve further inspection.

Relative_clause errors (144 instances), which require understanding of more complex sentence structures which involve main and subordinate clauses, prove to be particularly challenging: the model achieves a recall of just 0.17. This reflects its difficulty in capturing hierarchical syntax in probably longer sentences.

Other functionally important categories, including *Prepositions* (0.32), *Determiners* (0.33), and *Ref_device* (0.18), show similarly low recall, likely reflecting the model's limited syntactic generalization.

In sum, while the model's strong spelling performance is consistent across both FCE and REALEC, its recall drastically declines for more complex, abstract, or contextually subtle categories. Agreement and structural errors, in particular, remain problematic and require further investigation.

Error Code	**Gold**	**TP**	**FN**	**Recall**
Adj_as_collective	1	1	0	1.00
Vocabulary	1	1	0	1.00
Adverbs	1	1	0	1.00
Spelling	1503	1268	235	0.84
Countable_uncountable	9	7	2	0.78
Agreement_errors	184	124	60	0.67
Numerals	52	29	23	0.56
Capitalisation	77	41	36	0.53
Lack_par_constr	6	3	3	0.50
Adjectives	6	3	3	0.50
suggestion	18	9	9	0.50
Infinitive_constr	8	4	4	0.50
Tense_form	74	36	38	0.49
Noun_number	206	101	105	0.49
Category_confusion	122	59	63	0.48

Error Code	Gold	TP	FN	Recall
Articles	851	381	470	0.45
Formational_affixes	32	14	18	0.44
Possessive	35	14	21	0.40
note	5	2	3	0.40
Nouns	5	2	3	0.40
Pronouns	23	9	14	0.39
Derivation	8	3	5	0.38
Absence_comp_sent	215	49	166	0.23
lex_part_choice	159	36	123	0.23
Participial_constr	40	9	31	0.23
lex_item_choice	782	187	595	0.24
Verb_pattern	97	19	78	0.20
Noun_inf	5	1	4	0.20
Linking_device	74	14	60	0.19
Modals	26	5	21	0.19
Prepositional_noun	32	6	26	0.19
Tense_choice	368	66	302	0.18
Ref_device	114	21	93	0.18
Compound_word	23	4	19	0.17
Comparative_constr	19	3	16	0.16
Prepositional_adjective	13	2	11	0.15
Word_choice	227	34	193	0.15
Absence_explanation	177	26	151	0.15
Redundant_comp	117	18	99	0.15
Negation	22	3	19	0.14
Quantifiers	7	1	6	0.14
UNKNOWN	455	58	397	0.13
Conjunctions	32	4	28	0.12
Word_order	507	56	451	0.11
Inappropriate_register	105	10	95	0.10
Coherence	52	3	49	0.06
Relative_clause	144	8	136	0.06
Punctuation	644	28	616	0.04
Verbs	1	0	1	0.00
Discourse	7	0	7	0.00
Comparison_degree	5	0	5	0.00
Prepositional_adv	1	0	1	0.00
TOTAL	**8087**	**2909**	**5178**	**0.36**

Table 4.12: Recall per error type for XLM-RoBERTa on REALEC dataset (sorted by recall)

Analysis of Model Recall Performance on MERLIN

To conclude this preliminary quantitative analysis of the model's performance on the Italian dataset, MERLIN, XLM-RoBERTa achieves an overall recall of 0.53, which is higher than its performance on the English counterparts. This indicates that the model is able to capture slightly more than half of the annotated errors. However, a closer look at individual error types reveals substantial variability in performance, suggesting that some categories are considerably more challenging than others.

Some error types stand out for their relatively high recall. Categories such as *Coherence_Content-Jump* and *Grammar_Reflexive-Pronoun*, along with several other low-frequency types containing only one to four instances, achieve perfect or near-perfect recall. However, as previously noted in relation to FCE and REALEC, such limited sample sizes affect the

reliability of these results.

Among high-frequency categories, *Orthography_Grapheme* (154 instances), corresponding to spelling errors, and *Grammar_Preposition* (82 instances) perform relatively well, with recall scores of 0.70 and 0.60, respectively. These results suggest that the model, even for Italian, handles surface-level errors with reasonable robustness.

On the other hand, several categories expose clear limitations in the model's ability to generalize. For instance, *Grammar_Verb* (37 instances) and *Grammar_Word-Order* (51 instances) both yield recall scores below 0.25, highlighting persistent difficulties in processing syntactic structures. Similarly, *Grammar_Article* (72 instances) and *Grammar_Part-Of-Speech* (22 instances) show relatively low recall at 0.42 and 0.45, respectively, despite being common and structurally constrained phenomena in learner corpora. Notably, *Grammar_Agreement* (43 instances) also performs poorly, with a recall of 0.35. This suggests that the model struggles to capture subject-verb or noun-adjective agreement patterns.

Overall, although XLM-RoBERTa performs well in detecting surface-level orthographic errors, such as those involving spelling, capitalization, or punctuation, its recall drops significantly for more abstract linguistic phenomena. These include morphosyntactic errors like verb valency mismatches, clitic constructions, and article omissions, all of which are very frequent in Italian learners.

The results highlight XLM-RoBERTa's capacity to handle a wide range of error types in Italian, particularly when those errors are frequent and orthographically salient. However, its performance remains limited for structurally or semantically complex categories. Categories such as *Grammar_Word-Order*, *Grammar_Verb*, *Grammar_Agreement*, and *Coherence_Connector-Accuracy* remain particularly challenging.

Error Type	**Gold**	**TP**	**FN**	**Recall**
Coherence_Content-Jump	1	1	0	1.00
Orthography_Capitalization	43	36	7	0.84
Grammar_Reflexive-Pronoun	4	3	1	0.75
Orthography_Grapheme	154	108	46	0.70
Orthography_Word-Boundary	10	7	3	0.70
Sociolin_Variation	3	2	1	0.67
Grammar_Verb-Formation	17	11	6	0.65
Orthography_Punctuation	62	38	24	0.61
Grammar_Preposition	82	49	33	0.60
Grammar_Main-Verb	28	16	12	0.57
Grammar_Wrong-Inflection	52	29	23	0.56
UNKNOWN	295	162	133	0.55
Grammar_Inexistent-Inflection	26	14	12	0.54
Sociolin_Text-Type-Specificity	**17**	**9**	**8**	**0.53**
Grammar_Clitic	14	7	7	0.50
Pragmatics_Request	4	2	2	0.50
Intelligibility_Sentence	2	1	1	0.50
Orthography_Apostrophe	4	2	2	0.50
Grammar_Article	**72**	**30**	**42**	**0.42**
Grammar_Part-Of-Speech	22	10	12	0.45
Grammar_Conjunction	28	11	17	0.39
Intelligibility_Text	79	31	48	0.39
Grammar_Verb-Valency	**28**	**10**	**18**	**0.36**
Grammar_Agreement	**15**	**5**	**10**	**0.33**
V_Formulaic-Sequence-Form	3	1	2	0.33
Grammar_Negation	4	1	3	0.25
Grammar_Word-Order	**51**	**11**	**40**	**0.22**
Grammar_Verb	**37**	**5**	**32**	**0.14**
Vocabulary_Formulaic-Sequence	13	4	9	0.31
Vocabulary_Word-FS-Denotation	1	0	1	0.00
Coherence_Connector-Accuracy	1	0	1	0.00
TOTAL	**1172**	**616**	**556**	**0.53**

Table 4.13: Recall per error type for XLM-RoBERTa on the MERLIN dataset (sorted by recall)

Chapter 5

Error Analysis

This chapter presents a detailed error analysis based on the classification results reported in Table 4.11 and Table 4.13. The analysis is divided by language, treating English and Italian separately, and focuses on a selected set of error types that span syntactic, morphological, and sociolinguistic domains. These error types were chosen due to their prevalence, linguistic relevance[1], and partial alignment across the two languages.

For English, only the results from the FCE dataset are reported. Since both REALEC and FCE were evaluated using the same model configuration and represent the same target language, comparable patterns are expected. Thus, the FCE data is considered sufficiently representative for this purpose.

Each error type is examined through two perspectives. The first part focuses on the predictions made by the XLM-RoBERTa model, with the aim of identifying whether the predictions suggest plausible alternative correction strategies or whether discrepancies with the gold labels may point to annotation inconsistencies. The second part shifts attention to the generative outputs produced by Qwen, analyzing how the results presented in Table 5.1 and Table 5.2 manifest in individual examples.

To gain a more nuanced understanding of XLM-RoBERTa's performance, each error type is examined through three distinct types of misclassifications, as mentioned in Chapter 3.4. Together, these perspectives provide a more comprehensive view of model behavior, revealing patterns in correction strategies and the linguistic features associated with both accurate and erroneous predictions.

For both the classification outcomes and the model-generated corrections, I used the same set of sentences. For English, when an error type contained more than 50 sentences, I randomly selected approximately 50 sentences, supplemented with additional cases I had previously identified as particularly relevant for the study. When an error type contained fewer than 50 sentences, I analyzed all available sentences. For Italian, I analyzed all sentences for each error type, as the numbers were considerably smaller than for English. The same approach was applied to sentences containing only false positives and correct predictions: for English, I selected an average of 50 sentences, while for Italian, I used all 42 available sentences.

The files containing the classification labels used for the discriminative model's output and the generative model's output can be found in the project repository.[2] All the statistics and tables presented in this chapter (see Tables 5.1, 5.2, 5.3, 5.4, 5.5, 5.6, 5.7) are based on these manually labeled sentences.

[1] The extent to which an error type is significant for understanding how a language works, particularly in cases where grammatical ambiguity makes classification less straightforward

[2] Available at classified_sentences_fce.tsv, classified_sentencesFP_fce.tsv, classified_sentences_merlin.tsv, and classified_sentences_merlin_FP.tsv.

5.1 Preliminary Evaluation of Generated Corrections

The following tables provide an overview of Qwen's performance when generating grammatical corrections for English (see Table 5.1) and Italian (see Table 5.2). The criteria used for the classification have been previously presented in Chapter 3.4. For each error type, the total number of analyzed sentences (**Sent.**), generated outputs (**Gen.**), and their qualitative evaluation are reported (**Fully**, **Part. +**, **Part. -**, **No Mod.**, **Intro. Err.**, **Diverg.**). Boldface indicates the best-performing category, determined by the highest combined count of fully correct generations and partially correct generations that addressed the targeted error [3]. Italics mark the weakest category, identified by the lowest proportion of fully correct generations relative to the total number of generations.

Error Type	Sent.	Gen.	Fully	Part. +	Part. -	No Mod.	Intro. Err.	Diverg.
Verb Tense	54	265	46 (17.4%)	21 (7.9%)	47 (17.7%)	9 (3.4%)	20 (7.6%)	122 (46.0%)
Verb Agreement	13	60	2 (3.3%)	19 (31.7%)	0 (0.0%)	1 (1.7%)	7 (11.6%)	31 (51.7%)
Noun Agreement	**27**	**132**	**27 (20.5%)**	**31 (23.5%)**	**4 (3.0%)**	**2 (1.5%)**	**5 (3.8%)**	**63 (47.7%)**
Replacing Verb	51	245	43 (17.5%)	18 (7.3%)	29 (11.9%)	13 (5.3%)	9 (3.7%)	133 (54.3%)
Word Order	*50*	*243*	*28 (11.5%)*	*26 (10.7%)*	*41 (16.9%)*	*14 (5.8%)*	*14 (5.8%)*	*120 (49.3%)*
Total	195	945	146 (15.5%)	115 (12.2%)	121 (12.8%)	39 (4.1%)	55 (5.8%)	469 (49.6%)

Table 5.1: Qwen-generated sentence classification by error type and correction outcome for English (FCE), using annotation labels.

Table 5.1 reports Qwen's performance on English grammatical error types from the FCE dataset. Only these error types are considered for generation to ensure consistency with those examined in the error analysis. The model performed best on *Noun Agreement*, where 27 generations were fully correct and an additional 31 were partially correct while still addressing the intended error type, resulting in over 44% of generations being at least partially successful. In contrast, the model struggled the most with *Word Order* errors, producing only 28 fully correct generations and 26 partially correct ones out of 243, while nearly half of the outputs (120) resulted in divergent meaning. *Verb Tense* and *Replacing Verb* also showed relatively strong results in terms of fully correct outputs (46 and 43 respectively), though divergent meanings are still a large proportion of the generations (122 and 133). Performance on *Verb Agreement* was limited, with only 2 fully correct outputs and 19 partially correct, but a high proportion of divergent meanings (31 out of 60).

Error Type	Sent.	Gen.	Fully	Part. +	Part. -	No Mod.	Intro. Err.	Diverg.
Grammar_Article	36	172	20 (11.6%)	10 (5.9%)	19 (11%)	12 (7%)	7 (4.1%)	104 (60.4%)
Grammar_Agreement	**9**	**43**	**0 (0%)**	**10 (23.2%)**	**0 (0%)**	**3 (7%)**	**0 (0%)**	**30 (69.8%)**
Grammar_Verb	25	117	12 (10.2%)	11 (9.4%)	10 (8.6%)	10 (8.6%)	0 (0%)	74 (63.2%)
Grammar_Valency	17	83	6 (7.2%)	8(9.7%)	10 (12%)	4 (4.9%)	1 (1.2%)	54 (65%)
Grammar_Word-Order	*17*	*85*	*8 (9.5%)*	*0 (0%)*	*4 (4.7%)*	*11 (12.9%)*	*0 (0%)*	*62 (72.9%)*
Total	104	500	46 (9.2%)	39 (7.8%)	43 (8.6%)	40 (8%)	8 (1.6%)	324 (64.8%)

Table 5.2: Qwen-generated sentence classification by error type and correction outcome for Italian (MERLIN), using annotation labels.

For Italian, from Table 5.2, it can be seen that the model performs best on the *Grammar_Agreement* category, with 10 partially correct outputs out of 43 generations (23.2%), the highest relative proportion among error types. However, no generation was fully correct. The most challenging category appears to be *Grammar_Word-Order*, where only 8 out of 85 generations (9.5%) were fully correct and none were partially correct; the vast majority (62) were

[3]It is important to note that the model was provided only with the original sentences containing errors, without any indication of the error location or type. The 'targeted error' classification reflects the specific error types under analysis to evaluate whether and how the model addressed each category.

divergent. This indicates that Qwen struggles to successfully reorder sentence constituents in Italian, possibly due to the flexible word order of the language. Performance on *Grammar_Verb* and *Grammar_Valency* shows a modest proportion of both fully correct and partially + correct generations, indicating that Qwen often identifies the relevant area of the error but frequently fails to produce fully grammatical corrections. Overall, Qwen demonstrates cross-linguistic tendencies, such as stronger performance on errors involving article use and surface-level corrections, but its ability to handle deeper syntactic phenomena like agreement and word order appears limited, with performance in Italian overall being less robust than in English.

As shown in Table 5.3, the classification used for false positives slightly differs from that used in the error-type-specific analysis. In this case, no distinction was made between partially correct + and partially incorrect − generations; instead, a single 'partially' category was used. This is because the model predictions and generations could not be mapped to a specific error type. Rather than targeting a particular error, the goal here was to assess whether the model's correction was compatible with its classification, i.e., whether the model correctly identified a sentence as erroneous and generated a plausible correction.

Error Type	Sent.	Gen.	Fully	Part.	No Mod.	Intro. Err.	Diverg.
False Positives (FCE)	42	247	74 (30%)	18 (7.3%)	4 (1.6%)	6 (2.4%)	145 (58.7%)
False Positives (MERLIN	42	200	51 (25.5%)	21 (10.5%)	6 (3%)	0 (0%)	122 (61%)

Table 5.3: Qwen-generated sentence classification for English (FCE) and Italian (MERLIN), using annotation labels.

The results show the model's strength in handling false positives, particularly in English, where 30% of generations were fully correct. However, the high number of divergent generations in both languages suggests that the model often over-corrects or introduces irrelevant edits, especially in sentences that were already grammatically correct.

These results should be interpreted in light of this study's objective: the aim is not to perform automated correction or detection, but rather to support human evaluation. The model is intended as a tool to surface a range of plausible corrections, helping users reflect on linguistic ambiguity and visualize different ways an error could be interpreted or resolved. As such, generating as many 'good' corrections as possible is more important than optimizing a single correct output.

5.2 English

Table 5.4 groups model predictions, divided by error type, into the three categories previously mentioned in Chapter 3.4: *plausible correction*, *grammatical sentence*, and *model wrong*.

Error Type	Sentences	Model Correct: Plausible Correction	Model Correct Grammatical Sentence	Model Wrong
Verb Agreement	13	2 (11.5%)	2 (36.0%)	9 (52.5%)
Noun Agreement	27	3 (11.1%)	14 (51.9%)	10 (37%)
Verb Tense	55	5 (9.1%)	21 (38.2%)	29 (52.7%)
Replacing Verb	53	2 (3.8%)	32 (60.3%)	19 (35.9%)
Word Order	52	8 (15.4%)	21 (40.4%)	23 (44.2%)
Total	200	20 (10%)	91 (45.5%)	89 (44.5%)

Table 5.4: Distribution of XLM-RoBERTa predictions by error type, grouped by evaluation category (FCE): plausible correction, grammatical sentence, or incorrect output.

The model demonstrates the highest accuracy in cases involving verb replacement errors, where 60.3% of predictions are classified as correct due to the sentences being grammatically

valid, suggesting that many instances annotated as errors may represent acceptable linguistic variations. Conversely, verb agreement shows the poorest performance, with 52.5% of predictions being incorrect, indicating systematic difficulties in detecting subject-verb agreement violations. Across all error types, the model's predictions are almost evenly split between being wrong (44.5%) and identifying grammatically correct sentences (45.5%), with only 10% representing plausible alternative corrections. This distribution suggests that a significant portion of the annotated errors may involve subjective judgments or acceptable linguistic variants rather than clear-cut grammatical violations.

Notation In the example sentences that follow, tokens marked as incorrect in the gold standard are shown in bold (**word**), while tokens flagged by the model are enclosed in double braces ({{word}}). When the model correctly identifies a gold-labeled error, the token appears in bold within double braces (**{{word}}**). In the case of generated corrections, tokens modified by the model appear underlined (word). This convention enables a clear and concise comparison between the reference annotations, XLM-RoBERTa's predictions, and Qwen's generations. Only the error types under analysis are marked; other errors are omitted from the marking system for clarity.

5.2.1 Verb Agreement Errors

The FCE dataset includes 13 sentences containing verb agreement errors: 10 of these contain only false negatives and correct predictions, while 3 include both false positives and false negatives. Given the overall size of the development set, this represents a remarkably low frequency of such errors, indicating highly satisfactory model performance on this error type. Verb agreement errors involve mismatches between the subject and the verb form, often influenced by factors such as pronoun selection (see Example (3)), coordination of noun phrases (Example (4)), or cases where the annotator, maybe mistakenly, labeled the tokens as incorrect (Example (2)).

Moreover, since multiple acceptable corrections are plausible for this error type, it is worth examining whether the few residual cases, where the model's predictions do not fully align with the gold labels, can provide insight into the nature of these mismatches.

As shown in Table 5.4, across these 13 sentences, the model's outputs diverge from the gold annotations in several ways: in 2 cases, the model proposes alternative plausible corrections and is therefore not necessarily incorrect; in 2 additional cases, the sentences appear grammatical with respect to the specific error type analyzed; and in the remaining 9 cases, the model makes incorrect predictions.

(1) In other hand you ask me I have the chance for to choose two activities while I am at the camp. But is no easy, because {{all}} **is** very interesting for me.

(2) Moreover there **was** no discount **ticket**.

(3) Fortunately today everyone can take an aeroplane and go to America, Australia or wherever else **he wants**.

(4) The area for parking, space for play grown to the children or bus station near the mall **is** not really appropiate.

A particularly illustrative example of a sentence in which the model seem to suggest an alternative error location is sentence (1), which contains both false positives and false negatives. This sentence reveals an interesting case of interpretive flexibility in the model's treatment of subject–verb agreement. The gold annotation flags the verb 'is' in the phrase 'all is very interesting' as incorrect, based on a straightforward reading in which the plural subject 'all' requires the plural verb 'are'. The model, however, marks the noun 'all' as erroneous instead. Rather than considering this prediction entirely erroneous, it is worth acknowledging that this behavior may reflect the model's recognition that an agreement problem exists, even though with a different

hypothesis about its origin. Specifically, the model seems to assume that the subject should have been singular (e.g., 'everything is interesting') rather than requiring a plural verb.

Sentence (2) is an example of a grammatical sentence case. The model does not label any specific token as incorrect, as each token in its local context appears to be grammatically acceptable. These mismatches often arise due to a lack of broader contextual information. In this instance, it is likely that the annotator's decision to treat both the verb and the nominal subject as plural, namely, 'were' and 'tickets', was influenced by surrounding discourse. Therefore, this case should not necessarily be interpreted as a misclassification by the model.

Sentences (3) and (4) illustrate cases in which the model's predictions are clearly incorrect. Sentence (3) contains a mismatch between the subject, 'everyone', and the pronoun-verb pair 'he wants'. While 'everyone' is grammatically singular, it is often interpreted as semantically plural, which may explain the annotator's correction to 'they want'. However, the core issue may not lie solely in pronoun-number agreement. Rather, it involves the verb 'want' being used intransitively, which is ungrammatical in this context. The verb 'want' is strictly transitive and typically requires an explicit object (e.g., 'he wants to go'). The model failed to flag this error, possibly due to limited sensitivity to such syntactic constraints, or because this may be interpreted as a case of ellipsis. This highlights the complexity of detecting grammatical errors that intersect with both agreement and verb argument structure.

Sentence (4) illustrates a complex subject where multiple coordinated noun phrases form a compound subject. Learners often mistakenly use a singular verb, overlooking the need for plural agreement. The model fails to flag such errors, likely because its shallow parsing does not fully capture the subject's internal coordination structure. In this example, the annotator replaces 'is' with 'are', which is necessary considering the coordinated phrases 'parking' and 'space for play grown' joined by a comma, forming a plural subject. While the conjunction 'or' typically signals a choice requiring singular agreement (e.g., 'he or she prefers...'), here 'or' connects two coordinated compound elements. This opposition thus results in a plural subject, which requires plural agreement.

Such cases highlight the limitations of categorical annotation schemes in capturing gradient or ambiguous interpretations, and suggest that the model is capable of handling grammatical problems with a certain degree of interpretive depth.

Generative Model Evaluation on Verbal Agreement Errors

As noticeable in Table 5.1, Qwen demonstrates a generally promising ability to generate corrections for subject-verb agreement errors, with notable strengths in identifying mismatches between singular and plural subjects and verbs in common constructions. One illustrative success case is shown in Sentence (3), where the model transforms 'everyone... he wants' into the grammatically aligned 'they want', correctly handling both the agreement and the pronoun gender mismatch. Similarly, in Sentence (5), the correction of 'Are there some possibility' to 'Are there some possibilities' and 'Is there any possibility' demonstrates a strong understanding of quantifier-verb agreement and noun countability. Moreover, correction (a) appears to align with the classification suggested by XLM-RoBERTa, whereas correction (b) corresponds more closely to the gold annotation and the human annotator's suggested correction.

(5) **Are** there **some** {{possibility}} to rent the golfshoes?
(a) Are there some possibilities to rent the golf shoes?
(b) Is there any possibility to rent golf shoes?

However, these strengths are less consistent in more structurally complex or ambiguous contexts. For example, in Sentence (1), the Qwen model shows partial success in correcting verbal agreement errors. It consistently attempts to repair the ungrammatical sequence 'is no easy' across multiple outputs, with several versions producing well-formed alternatives such as 'it is not easy' or 'this is no easy'. This suggests that the model recognizes the need for a subject to precede the verb and adjusts for standard syntax. However, the second related issue involving the phrase 'all is very interesting' remains uncorrected in all generated variants. This indicates

a blind spot in the model's handling of plural subjects followed by singular verbs, a pattern that was also observed in other examples (e.g., Sentence (4)). Overall, Qwen demonstrates sensitivity to verb agreement when the subject-verb relationship is syntactically close and clearly marked, but struggles with plural quantifiers and more globally distributed agreement constraints.

Finally, in relation to the above mentioned sentence(2), that could have mirrored different correction approaches, Qwen shows that both the options, the one found by the annotator which consider the nominal subjects and the predicate as plural and the learner's original sentence which involved the singular are plausible versions of the sentence.

With the data available, given the used prompt, Qwen's performance on verbal agreement errors is strongest in canonical structures and short-distance dependencies, especially where the subject-verb relationship is clearly marked and follows high-frequency patterns. Its limitations emerge in the presence of lexical ambiguity, long-distance dependencies, or competing syntactic interpretations. These findings indicate that while Qwen possesses a functional grasp of subject-verb agreement, its generative strategies are still influenced by surface-level heuristics and may not fully generalize to linguistically nuanced cases.

5.2.2 Noun Agreement Errors

As observed with verbal agreement errors, the model also fails to flag noun agreement errors annotated in the gold labels. I identified 27 sentences exhibiting this error, of which 3 cases involve divergent classifications that actually highlight alternative corrections, 14 cases show the model behaving reasonably because the sentence is grammatical, and 10 cases reflect incorrect predictions by the model (see Table 5.4).

(6) In my opinion, people will give importance to how their **personality** {{develop}} and they will give up worrying about what they wear.

(7) People who don't like shopping have their own reasons of {{this}} **situations** (maybe they had an accident with a rude shop-assistance, bought not fresh food one day and do not want to do that any more, had bad experience with other people in shops).

(8) In the letter you said that accommodation provided can be either **tents** or **log cabins**.

(9) She wears trousers like **men** and green flowery blouses.

(10) I hope you find my **opinion** useful.

(11) For all these **reason**, I want to ask for my money back and I hope to hear from you soon.

(12) Also, it was certain to me to bought a discount **tickets** cause I am a student .

Instances involving both false positives and false negatives, such as sentences (6) and (7), illustrate alternative solutions to agreement errors and offer a more nuanced perspective. These cases often reveal the model's sensitivity to syntactic structure. Sometimes, the model correctly identifies an agreement mismatch and may simply identify a valid alternative that the gold annotations do not consider. A common pattern is the model flagging the verb instead of the noun; for example, in (6), where the gold label marks the noun 'personality' as incorrect, the model instead flags the verb 'develop'. This suggests an underlying awareness of agreement constraints. A similar pattern occurs in sentence (7), where the model flags the demonstrative pronoun 'this' rather than the substantive 'reasons'.

Sentences from (8) to (10) are grammatical, yet the gold labels identify errors, likely due to contextual knowledge.

In sentence (8), the learner uses the plural 'tents' and 'log cabins', but the gold annotation expects singular forms. While the use of the singular might reflect a more formal or collective reading (e.g., 'tent accommodation'), the learner's sentence is fully grammatical in English.

The model treats the sentence as correct, and in this case, its behavior is arguably more aligned with fluent usage than the gold annotation.

Other cases, such as (9) and (10), show potential over-interpretation by annotators. In (9), the learner writes 'like men', where the gold label suggests the correct form is 'like a man'. However, the plural form is not ungrammatical, and the model's decision not to flag it may reflect semantic uncertainty rather than grammatical oversight.

In Sentence (10), the annotator marks the noun 'opinion' as incorrect, presumably favoring a plural form such as 'opinions' to better reflect the idea of multiple suggestions. However, this judgment may be influenced by prior context, perhaps earlier in the text, the learner had indeed shared several ideas. Taken in isolation, though, the sentence is grammatically correct. Thus, this instance highlights how gold labels can sometimes reflect discourse-level interpretations that are not strictly necessary from a syntactic persepective.

Taken together, these examples highlight the difficulty of evaluating model performance solely based on misalignment with annotated labels. While some false negatives clearly indicate model limitations, such as undetected number mismatches in quantifier-noun combinations, others reflect either grammatical variability or questionable annotations. In such cases, the model's 'errors' may in fact reflect appropriate grammatical knowledge.

Finally, two clear cases of misclassifications appear in (11) and (12). In sentence (11), the singular noun 'reason' follows the plural quantifier 'these'. The correct form should be 'these reasons', yet the model fails to detect the number mismatch between determiner and noun, resulting in a false negative. The same applies to 'tickets', which is preceded by the singular article 'a' and therefore requires a singular substantive.

Overall, XLM-RoBERTa shows a moderate degree of syntactic awareness in handling noun agreement errors, with its most common issue being the mislocalization of errors rather than a failure to detect them. At the same time, these examples call into question the limits and potential overreach of the current annotation scheme.

Generative Model Evaluation on Noun Agreement Errors

As shown in Table 5.1, noun agreement errors account for a substantial portion of Qwen's successful corrections on the FCE dataset, with 44% of all noun agreement errors being correctly identified and corrected.

In several of the sentences previously discussed, Qwen's behavior reveals an interesting pattern: its generations sometimes align with XLM-RoBERTa's classifications and other times with the gold annotations.

For instance, in sentence (13), Qwen shows variable behavior: in correction (a) it generates 'their lives,' correcting the number mismatch and thereby agreeing with the gold standard, while in correction (b) it leaves 'their life' unchanged, effectively aligning with XLM-RoBERTa's classification that treated this phrase as acceptable [4].

(13) Since the electric fire and microwave oven had been invented, their **life** has been far easier than before.
(a) Since the electric fire and microwave oven had been invented, their lives have been far easier than before.
(b) Since electric fires and microwave ovens had been invented, their life has been far easier than before.

Similarly, in examples such as (11), where XLM-RoBERTa accepted plural forms like 'reasons' as correct, Qwen also tends to preserve these variants in its generated outputs, suggesting that it, too, treats them as grammatical despite their divergence from the gold standard.

[4] While the plural agreement between possesive and noun is grammatically valid, learners often default to the unmarked singular life when generalizing about human experience, and the model does not flag the form. This may reflect not a lack of grammatical sensitivity but a tolerance for learner variation.

Overall, Qwen's output tends to agree with the classifier in cases where grammaticality is defensible and diverges when more natural or semantically aligned corrections are possible. This suggests that both systems exhibit a relatively permissive interpretation of plural/singular variants.

5.2.3 Tense-related Errors

The analysis of model behavior on tense-related errors reveals several patterns, highlighting both annotation inconsistencies in the data and the model's limitations due to lack of contextual information. The model fails to correctly classify this error 162 times, distributed across 125 sentences. I manually analyzed 55 sentences. In 5 cases, although the model's predictions do not align with the gold labels, they suggest plausible alternative corrections. In 21 cases, the model's outputs are grammatically correct in isolation, yet the sentences are labeled as incorrect in the dataset due to tense inconsistencies with the surrounding discourse, context to which the model does not have access. The remaining 29 sentences represent genuine misclassifications by the model (see Table 5.4).

For instance:

(14) I told her everything abat my boyfreind and when we {{were met}} each other and evertauly we **were falling** in love.

(15) So I hope I **{{could}} help** you.

(16) Because I **did** it for the first time.

(17) There **are** Robert, Alan and Jack Ruby instead of them.

(18) We **became** passive people, we follow the same routine every day and we live in a society where rules are strongly fixed.

(19) When the concert started I was preparing some drinks for the band because when the concert finished they **were** very tired and tersty.

(20) It's happened three months ago, and during this months I **am** very upset.

Sentence (14) and sentence (15) are cases of plausible model classifications. The first may be better understood as a case of mislabeling in the dataset, while the second reflects a alternative interpretation of the error at hand. The phrase 'were met', in sentence (14), is labeled as correct in the gold data, even though it is clearly ungrammatical. The model correctly flags it as an error which aligns with the expected structure ('we met'). However, it fails to flag 'were falling in love', whose correct form should have been the simple past 'fell in love', which better suits the narrative context. Despite the first accurate judgment, the model's output is marked as a false positive due to incorrect gold labels.

In sentence (15), while the learner uses 'could', which typically expresses past or hypothetical ability, the intended meaning is to express a completed action with relevance to the present, appropriately conveyed by the present perfect 'have helped'. Interestingly, the model flags only 'could' as incorrect, which may suggest the use of 'can' instead. This substitution implies simultaneity between the act of helping and the present, rather than posterior completion. The model's partial flagging again underscores its limitations in disambiguating modality, time reference, and aspect when evaluating grammatical correctness.

Sentences from (16) to (19) contain verb tense choices that are labeled as incorrect in the gold annotations, even though they are grammatically acceptable.

In sentence (16), the gold correction from FCE is 'was doing', marking the use of simple past as inappropriate. However, this correction only becomes meaningful when viewed in the broader discourse context, such as in a contrastive or progressive narrative. Without the preceding sentence, the model correctly interprets the simple past as valid. This phenomenon frequently

occurs in short, contextually isolated sentences that contain only one main predicate, making the intended tense relation difficult to infer.

A similar problem arises in the sentence (17), where the use of the present tense verb 'are' is labeled as incorrect, although no temporal cues in the sentence indicate a preference for a different tense. Again, this points to the need for discourse-level information to evaluate tense appropriateness reliably.

Tense distinctions between the past simple and present perfect remain particularly challenging. In sentence (18), the past simple verb 'became' is not strictly incorrect, but the suggested use of the present perfect ('have become') could better convey the ongoing impact of that change. The model does not detect this more subtle pragmatic nuance.

In other cases, see sentence (19), the difficulty does not stem from overt grammatical errors, but from the learner's tendency to default to expressions of contemporaneity even when they likely intend to convey anteriority or posteriority. These constructions are grammatically correct, and the model, lacking access to broader discourse context, has no way of detecting the underlying pragmatic mismatch. As a result, it does not flag these sentences, even though a more nuanced temporal structure would better align with the intended meaning.

Although the sentence is grammatical and passes unnoticed by the model, the temporal relation it expresses is pragmatically ambiguous. The learner appears to describe the band as already tired and thirsty at the time drinks were being prepared. In contrast, the correction, 'would be', assumes a predictive relation: the speaker anticipates their future state. This shift is also marked in the correction by the use of 'after the concert'. The model, however, does not pick up on this aspectual nuance, highlighting the difficulty of identifying discourse-level temporal dependencies in otherwise well-formed sentences.

In contrast, there are cases in which the model fails to flag genuinely incorrect tense usage, even when provided with clear intra-sentential cues. In sentence (20), the main clause indicates a past event, yet the model does not flag the present tense 'am' as inappropriate, missing the requirement for present perfect, 'have been', to express a continuing state.

Generative Model Evaluation on Tense-related Errors

As indicated in Table 5.1, verb-related grammatical errors show a lower correction rate overall, with only 10.2% of errors being fully corrected and 63.2% remaining uncorrected, suggesting that tense and aspect errors pose a greater challenge for Qwen.

The corrections generated by Qwen for sentences annotated with tens-related errors exhibit a notable range of surface fluency and lexical variation. However, despite producing grammatically well-formed outputs in most cases, it does not consistently resolve deeper issues related to tense, aspect, or temporal sequencing. Three major patterns can be identified from the analysis of Qwen's responses to the sentences discussed earlier.

First, it frequently preserves the original tense used by the learner, particularly in cases where the correction involves a shift from past simple to present perfect or past perfect. For instance, in Example (18), the gold correction replaces 'became' with 'have become' to express recency and relevance to the present. None of Qwen's outputs suggests this correction; instead, they maintain the past tense or introduce minor stylistic changes (e.g., modifying adjectives or punctuation). This case shows that while the generated outputs are grammatically correct, Qwen tends to mirror the surface structure of the input rather than resolve subtle aspectual mismatches.

Second, there are instances where Qwen demonstrates partial sensitivity to tense, modality, and aspect, offering multiple plausible alternatives. In Example (15), where the gold correction replaces 'could help' with 'have helped', Qwen produces a range of tense forms, including 'can help', 'could have helped', and 'hope to help'. These variants reflect different aspectual readings (e.g., contemporaneity vs. anteriority), although only one approximates the intended correction.

Finally, Qwen struggles with cases where the learner's use of tense is grammatical but contextually inadequate, often defaulting to contemporaneity. In Example (19), the sentence implies a future state of tiredness and thirst following a concert. The gold correction introduces a modal ('would be') and a temporal marker ('after the concert') to clarify the sequencing.

However, all Qwen variants retain the original structure, missing the opportunity to express posterior temporal relations. A similar issue arises in Example (20), where the model fails to substitute 'am' with 'have been', despite the discourse context requiring a present perfect form to describe an ongoing emotional state since a past event.

These findings mirror the behavior observed in the classification model: both systems show a preference for surface-level fluency and grammaticality, while overlooking deeper pragmatic or discourse-level tense relations. Moreover, in cases where gold annotations are inconsistent, such as the ungrammatical phrase 'were met' in Example (14), Qwen often aligns more closely with grammatical intuition than with the provided labels.

5.2.4 Replacing Verbs

In the category of verb rewriting errors, XLM-RoBERTa demonstrates consistently poor performance, as shown in Table 4.11. Qualitative inspection reveals that the model struggles not only with identifying inappropriate or incorrect verb forms, but also with recognizing that an issue exists in the first place, particularly when the error involves sociolinguistic nuance, stylistic appropriateness, or pragmatic adequacy. While the model can occasionally handle more clear-cut grammatical violations, it often fails to capture subtler errors that require a deeper understanding of idiomaticity, formality, or register.

This trend is largely explained by the nature of the data (see Table 5.4): of the 53 analyzed sentences[5] in which this error type occurs, 32 involve grammatically correct sentences where the gold annotations target subtle pragmatic or stylistic issues. Only 19 sentences represent clearly ungrammatical constructions, even though the intended meaning remains understandable. The remaining 2 cases reflect alternative but plausible corrections suggested by the model.

(21) In the afternoon she had phoned at a Francis's mother and she **explained** {{her}} the secret, then she phoned at her best friend and she speak about that too; she phoned at many people all the day.

(22) I've **seen** your festival and I would like to give you my suggestions to it.

(23) Agatha said that it was the greatest present she could ever **get**!

(24) We **put off** the lights and then the door opened and I just heard 'What the hell is going on?!'

(25) So we can **do** shopping instead.

In sentence (21), the annotator suggests an alternative verb for 'explained', namely 'told', resulting in the corrected phrase 'told her the secret'. The model, however, flags the direct pronoun 'her', suggesting the insertion of a missing preposition 'to', leading to the construction 'explained to her'. This is a very plausible correction: while it does not fully resolve the fluency issue related to verb choice, it addresses the grammatical problem present in the original sentence.

Sentences (22) and (23) contain verb choices that, even though not optimal, result in sentences that are still grammatically correct and easily understandable. As a result, the model does not flag any errors.

Sentences (24) and (25) are cases in which the incorrect use of a verb results in ungrammaticality. In sentence (24), the model fails to detect that the verb phrase 'put off the lights' is not idiomatic; 'put off' typically means to postpone, whereas the appropriate expression here would be 'turn off'. Similarly, in sentence (25), the model does not flag the non-native construction 'do shopping'.

Overall, these findings suggest that XLM-RoBERTa's understanding of verb semantics is largely surface-level, with minimal sensitivity to collocational norms, contextual adequacy, or sociolinguistic appropriateness.

[5]The total number of sentences containing mispredictions for this error class amounts to 140.

Generative Model Evaluation on Replacing Verb Errors

As shown in Table 5.1, verb replacement errors often go uncorrected by Qwen, with more than half (54.3%) of such instances left unchanged, suggesting a tendency to overlook contextually inappropriate verb choices even in cases where errors were clearly not contextually adequate. For instance:

(26) According to the advertisement you **gave**, there have been some important differents.
(a) According to the advertisement you gave, there have been some important differences.

(27) But when we talked to the doctor we felt more relaxed because he **answered** us that the operation was a very simple one.
(a) But when we spoke to the doctor we felt more relaxed because he told us that the operation was a very simple one.

(28) You can not **lose** the time because all the other shops could be closed when you arrive.
You can not waste time because all other shops could be closed when you arrive.

In sentence (26), the model correctly revised the noun 'differents' to 'differences', yet failed to address the less idiomatic verb 'gave', which should have been replaced with 'published' to align with the formal register.

Nevertheless, there are examples where the model did succeed in generating corrections that matched or even improved upon the gold standard. In sentence (27), the model correctly replaced both 'talked' with the more natural verb 'spoke', and 'answered us' with 'told us', aligning closely with idiomatic usage.

Similarly, in sentence (28), the model demonstrates sensitivity to lexical choice and collocational appropriateness by replacing 'lose the time' with 'waste time', a more idiomatic expression in this context.:

These examples indicate that while Qwen often defaults to conservative strategies, it can occasionally generate semantically and stylistically appropriate verb corrections. However, its performance remains inconsistent, particularly when subtle shifts in verb semantics or collocational patterns are required.

5.2.5 Word Order Errors

For errors related to word order, XLM-RoBERTa exhibits some interesting and consistent patterns. Out of the 228 token-level word order errors identified in the dataset, spanning 79 sentences, 66 of these sentences contain only false negatives and correct predictions, while 13 contain only false positives. This distribution suggests that the model tends to overlook such errors rather than falsely predict them, highlighting its limited sensitivity to structural rearrangements.

Out of the total 79 sentences, I manually examined 52: 39 with only false negatives or correct predictions, and 13 with only false positives. Within the examined set, 8 cases appear to suggest an alternative but still correct classification; 21 cases, although less fluent, involve sentences where the model's prediction diverges from the gold label since the sentence is grammatically acceptable; and 23 cases represent genuine misclassifications by the model (see Table 5.4).

(29) I **think {{also}}** that it will be a good idea to take some general plans of the school area.

(30) It was a **Friday {{suny}}** afternoon, Scala was really excited to tell her the news.

(31) Nowadays, people **seem to be always** worried.

(32) I am disappointed because of the horrible evening I have had, I was on holiday in London and I did not have **time enough** to spend there, but when you said that show

was the best I decided to go.

Sentence (29) is an example of a plausible prediction. The annotator labeled as incorrect both 'think' and 'also', while XLM-RoBERTa identifies as erroneous only 'also'. This pattern indicates that the model may pick up on the more conspicuous token, often the adverb, while failing to detect the incorrect placement of the co-dependent verb or auxiliary. The same issue arises with adjective-noun inversion. In the sentence (30), both 'Friday' and 'suny' are labeled as incorrect (misordered noun-adjective pair), while in the predictions only 'suny' is marked, which, most probably is due to the coexistent spelling error.

Sentence (31) can be considered grammatically correct, even though the annotator suggests a correction to 'always seem to be'. While the suggested version may sound more natural in written English due to typical adverb placement, the original ordering 'seem to be always' is also acceptable, especially in spoken or informal contexts. The variation reflects a matter of style and emphasis rather than strict grammaticality.

Finally, sentence (32) is a case of incorrect prediction by the model, which fails to identify the incorrect placement of the adverb 'enough' in that specific position. In standard usage, 'enough' typically follows the adjective or noun it modifies, so the correct phrasing should be 'enough time' rather than 'time enough'.

These examples reinforce the model's difficulty with syntactic coordination across short spans. Word order violations, by nature, involve dependencies between elements; however, the model appears to treat each token independently, thereby missing the broader structure of the error. This limitation may also reflect the simplicity of the annotation scheme, in which labels like i and c conflate distinct error types such as word order, missing words, and grammatical inaccuracies. I will return to this point in the discussion.

Overall, this partial recognition undermines the model's performance on word order issues and suggests a limitation in its internal representation of local syntactic relationships.

Generative Model Evaluation on Word Order Errors

As shown in Table 5.1, word order errors present a challenge for Qwen, with nearly half of the instances (49.3%) left uncorrected. The model shows only partial success in addressing this error type, with performance varying considerably depending on the syntactic construction.

(33) First of all, I would like to tell you that I've never had **a such** good time in my life.
(a) First of all, I would like to tell you that I've never had such a good time in my life.
(b) First of all, I would like to tell you that I have never had such a good time in my life.
(c) First of all, I would like to tell you that I've never had such good a time in my life!

(34) She was **a such** wonderful person in our class.
(a) She was such a wonderful person in our class.
(b) She was a very wonderful person in our class.
(c) She was a truly wonderful person in our class.

(35) I would **be also** grateful if you could send me information about the average living cost at California as I have to arrange the amount of money that I am going to spend , and some advice about the clothes that I need .
(a) I would also be grateful...

(36) Technical and Warm Home People 's way of life **has considerably been changed** by rapdily developing mordern technology.
(a) Technical and warm home people's ways of life have been changed considerably by rapidly developing modern technology.

(b) Technical and warm home people's ways of life have been considerably changed by rapidly developing modern technology.

(37) Most of the time **{{is}} shopping** fun and enjoyable.
(a) Most of the time is shopping fun and enjoyable.
(b) Most of the time, shopping is fun and enjoyable.
(c) Most of the time is spent shopping, which is fun and enjoyable.

(38) **This kind of problem you could avoid** when you look for big halls at the beginning.
(a) You could avoid this kind of problem when you look for big halls at the beginning.

While many generated corrections preserve the original (often incorrect) word order, the model occasionally produces accurate and idiomatic rephrasings. These successful cases typically occur once or twice across the five generated outputs per sentence, such as in the case of sentence (33), suggesting that effective corrections arise more from sampling diversity than from consistent grammatical modeling.

Qwen seems to perform visibly well in the way it handles mis-ordered noun phrase modifiers, by preserving and slightly modifying the original structure of the sentence and also by offering a different variety of corrected alternatives, as noticeable in sentence (34). While only the first option exactly matches the gold, the others demonstrate plausible reformulations that improve fluency and acceptability, even if they diverge slightly in meaning.

The model also tends to perform better when the word order involves adverbs modifying auxiliaries or main verbs, such as in sentences (35) and (36). In the sentence(35), Qwen successfully relocates the adverb to produce a more natural structure such as 'I would also be grateful...'. All the generated corrections match the gold correction in relation to the word order error highlighted. In sentence (36) the model produces two grammatically correct outputs, even though only the first one matches the gold correction, while the second one deviates from the annotators' corrections and shows alternative solutions to the problem.

In cases involving inversion between nominal subjects and predicates, Qwen produces mixed results (37). The model alternates between reproducing the ungrammatical inversion, correcting it and also introducing new erroneous versions of the sentence, demonstrating a form of semantic rephrasing beyond surface correction, even if incorrect.

Furthermore, Qwen handles anticipatory object constructions with surprising accuracy, as seen in sentence (38). Although only one of the 5 generated corrections actually fixes the word order issue (shown in sentence (38) a), restoring the canonical subject-verb-object order could have been challenging.

Overall, Qwen performs best on word order errors that involve auxiliary–adverb placement and local modifier reordering. Its ability to generate multiple variations increases the chance of producing a valid correction, but this success is inconsistent across syntactic structures. The model is less reliable in handling global sentence-level reorderings or subtle pragmatic inversions, which require deeper syntactic and discourse-level reasoning.

5.2.6 False Positives - FCE

In the case of the English FCE portion of the dataset, 129 sentences contain only false positives and correct predictions. Upon closer inspection of 51 of these sentences, 6 cases involve mispredictions that suggest plausible alternative corrections, 14 cases reveal clear errors in the gold labels, and 31 cases represent genuine model errors (see Table 5.5). This analysis calls into question the reliability of certain evaluation metrics that rely solely on confusion matrices.

Error Type	Sentences	Model Correct: Plausible Correction	Model Correct: Mislabeled in MultiGED	Model Wrong
False Positives	51	6 (11.8%)	14 (27.4%)	31 (60.8%)

Table 5.5: Distribution of XLM-RoBERTa false positive predictions, grouped by evaluation category (FCE): plausible correction, mislabeled in the MultiGED annotations, or incorrect prediction.

The following sentences provide examples of these trends:

(39) Second, I went to the theatre **{{exactly}}** {{seven}} thirty in the evening.

(40) Please let me know if you {{are agree}} with us!

(41) If the famous person doesn't mind {{in}} being filmed...

(42) I would like to complain about the show which you presented {{on last Saturday}}.

(43) Afterwards I decided to go for a meal in the theatre restaurant, but {{It}} was closed.

(44) Apologise {{me}} for {{disturb}}.

(45) Okay, I hope to have {{news}} from you very soon.

(46) I couldn't **{{consentrate}}** {{on}} the play any more .

Sentence (39) could be a case of plausible classification by the model. The gold annotation marks only 'exactly' as incorrect, likely expecting a correction such as 'at exactly seven thirty', considering that tokens following a missing word are labeled as incorrect in MultiGED. The model, however, flags both 'exactly' and 'seven' as incorrect. While this results in a false positive for 'seven', the model's prediction is still plausible. The expression 'exactly seven thirty' without a preposition is uncommon in standard English. By marking 'seven' as well, the model appears to treat the entire time expression as problematic, possibly detecting the absence of the required preposition 'at'. This suggests that the model is sensitive to phrase-level structures, even if its token-level predictions do not fully align with the gold annotations.

Sentences (40) to (44) illustrate cases in which the gold labels omit some errors. The incorrect use of 'are agree' in sentence (40) is not flagged by the gold labels, yet the model correctly identifies it. The same happens with incorrect use of prepositions, both in verbal constructions in sentence (41), and in nominal constructions as in sentence (42). In sentence (41), the verb 'mind' is typically followed by a complement clause in the form of a gerund phrase, such as 'being filmed', which functions as its direct object. Crucially, this complement is not introduced by any preposition. The construction is thus non-finite but syntactically complete, and any insertion of a preposition (e.g., 'mind in being filmed') results in an ungrammatical form. Moreover, in sentence (42), the preposition 'on' is not required here. The model captures this, while the annotation does not. Capitalisation, which will also happen for Italian, is also inconsistently annotated. For instance, in sentence (43) the capitalisation of 'It' mid-sentence is flagged by the model but not in the gold labels. Some sentences, such as sentence (44), are completely ungrammatical but still labeled as correct in the gold.

Sentence (45) and (46) are, on the other hand, true false positives, namely cases in which the model is mistakenly flagging a correct token.

These examples indicate that many false positives are not true errors but reflect valid model judgments. Inconsistencies in the gold annotations, particularly around spelling, article/preposition usage, subject–verb agreement, and capitalisation, mean that the model is penalised in evaluation despite making correct predictions. These findings support the need for more nuanced evaluation procedures that include qualitative error analysis and question the infallibility of gold-standard annotations.

Generative Model Evaluation on False Positives

An analysis of Qwen's behavior on false positive cases reveals a mix of conservative corrections, creative reformulations, and generally reliable handling of common learner errors (see Table 5.3).

In Example (40), the learner erroneously combines the auxiliary 'are' with the verb 'agree', resulting in the ungrammatical construction 'are agree'. Qwen offers a range of correction strategies. The first output is the most minimal and accurate, removing the auxiliary to yield 'if you agree with us', which closely preserves the original structure. In other generated outputs, the model either leaves the sentence unchanged, thus reproducing the error, or rephrases it using constructions such as 'are in agreement', which, while more formal, are also grammatically correct.

In contrast, in Example (41), the model consistently produces correct outputs, omitting the incorrect preposition 'in' and selecting the appropriate structure 'mind being filmed'. This suggests that Qwen handles certain verb-complement constructions involving non-finite clauses with high accuracy, particularly when no prepositional element is required.

More complex or severely ill-formed sentences, such as in Example (44), prompt the model to take more creative liberties. Corrections include both grammatical but semantically loose rewrites (e.g., 'Apologize to me for disturbing') and reduced constructions that are awkward or unidiomatic (e.g., 'Sorry for disturb'). In such cases, the model demonstrates sensitivity to the ill-formedness of the input but produces corrections that range in grammaticality and appropriateness.

Spelling errors are consistently and reliably corrected across outputs. For example, errors like 'morden' or 'combinient' are always replaced with the correct forms ('modern', 'convenient') without introducing unrelated modifications.

Finally, when the input sentence is already grammatical the model shows accurate behavior:

(47) {{All the time}} I {{am}} thinking about Peter.
(a) All the time I am thinking about Peter.
(b) Every moment I am thinking about Peter.

As in Example (47), Qwen tends to preserve the original phrasing (a) or offer slight paraphrases (b), maintaining both grammaticality and semantic equivalence. This conservative behavior suggests that the model, in the absence of clear errors, refrains from over-editing, which is desirable in learner-oriented applications.

5.3 Italian

As in the case of English, Table 5.6 presents the classification of the model's predictions for Italian sentences into three categories: *plausible correction*, *grammatical sentence*, and *wrong* prediction.

Error Type	Sentences	Model Correct: Plausible Correction	Model Correct: Grammatical Sentence	Model Wrong
Grammar_Article	36	3 (8.3%)	6 (16.7%)	27 (75%)
Grammar_Agreement	9	0 (0%)	3 (33%)	6 (67%)
Grammar_Verb	25	0 (0%)	6 (24%)	19 (76%)
Grammar_Valency	17	2 (11.7%)	5 (29.4%)	10 (58.9%)
Grammar_Word-Order	17	3 (17.6%)	6 (35.3%)	8 (47.1%)
Total	104	8 (7.7%)	26 (25%)	70(67.3%)

Table 5.6: Distribution of XLM-RoBERTa predictions by error type, grouped by evaluation category (MERLIN): plausible correction, grammatical sentence, or incorrect output.

It can be seen that XLM-RoBERTa's performance on the Italian MERLIN dataset is considerably weaker compared to the English FCE results. The model achieves correct predictions in only 32.7% of cases (combining plausible corrections and grammatical sentences), with a substantially higher error rate of 67.3%. Grammar articles present the greatest challenge, with 75% of predictions being incorrect, followed closely by grammar verbs at 76%. The model shows relatively better performance on word order errors (47.1% incorrect) and valency issues (58.9% incorrect). Notably, the proportion of plausible alternative corrections is slightly lower (7.7%) compared to the English dataset, indicating that the model's errors in Italian are less likely to represent valid linguistic alternatives.

The examples presented follow the same notation system described in Paragraph 5.2. However, it should be noted that each example begins with the original sentence in Italian, annotated using the notation system, followed by its English translation in *italics*. The translation reflects the intended meaning, specifically, the annotator's corrected version of the sentence.

5.3.1 Grammar_Article

Among all error categories, *Grammar_Article* emerges as one of the most frequent, with 42 occurrences distributed across 36 sentences in the dataset. These include incorrect, missing, or malformed articles, encompassing both simple forms (e.g., *il*, *la*, *un*) and contracted forms (e.g., *dell'*, *all'*, *l'*). A substantial subset of these errors involves *prepositional contractions* (i.e., *preposizioni articolate*, such as *di* + *il* → *del*), which are a common, though not always obligatory, feature of Italian grammar. The model struggles both to identify when such contractions are required and to flag their superfluous use.

I checked all 36 sentences and found that in 3 cases, the model likely proposed a different correction by labeling a different word as incorrect. In 6 cases, the sentences were actually grammatical, meaning the model's prediction, although different from the gold label, was still correct. In the remaining 27 cases, the model's predictions were simply wrong.

(48) Quanto costa {{il}} **questo** viaggio?
How much does this trip cost?

(49) La sera io posso andare {{al}} **un** concerto con la mia padre o andare in un ristorante con la mia famiglia
In the evening I can go to a concert with my dad or go to the restaurant with my family.

(50) perché non parli un po' **di** matrimonio quando telefoni con me?
Why don't you talk about your wedding when you call me?

(51) Penso che la societa sia formata così, anchio avevo quel pensiero ma quando sono andato **in** viaggio in Africa...
I think that society is made this way, I had that thought as well, but when I went on a trip to Africa...

(52) Se accettassi, dovrei registrare i fogli **di** ufficio, come no.
If I accepted, I would have to register the papers of the office, right?

(53) Qualce volta, vado **in** calcio.
Sometimes I go to football.

(54) È stata **una** esperienza straordinaria di attraversare **gli** Alpi.
It has been an amazing experience to cross the Alps.

(55) Dopo **15** ottobre sono costretta di andare dalla mia sorella Maria (Maria è malata).
After the 15th October I am forced to go to my sister Maria (Maria is sick).

(56) E come stanno tua sorella e **tuoi** genitori?
And how are your sister and your parents doing?

Examples (48) and (49) illustrate cases in which the model appears to propose an alternative correction by assigning the error label to a different token than the one annotated in the gold standard. In (48), for instance, the gold labels the demonstrative 'questo' (*this*) as incorrect, while the model instead flags the article 'il'. In (49), the gold correction addresses an article contraction error, where the ungrammatical sequence 'al un' (*to the*) can be corrected either to 'al' (*to the*), following the minimal target hypothesis, or to 'a un'(*to a*), as in the extended target hypothesis. Since the gold labels adopt the minimal target hypothesis, the token 'un' is marked as incorrect. The model, however, labels 'al', suggesting a correction aligned with the extended target hypothesis.

Sentences (50) and (51) represent cases in which the sentences appear to be grammatical, either due to misannotations in the gold labels or despite their somewhat unusual meaning. Sentence (51) includes a token labeled as a *Grammar_Article* error in the gold standard, specifically, the preposition 'in' in the phrase 'in viaggio', even though this expression is grammatically correct in Italian and does not require an article. In this case, the model's prediction was accurate, and the error seems to result from a misannotation in the gold. Sentence (50), on the other hand, contains a prepositional not so fluent use following the verb 'parli' (*you talk*). However, the sentence could still be interpreted meaningfully, though likely not in the way the learner intended. The use of 'del' (*of/about the*) in the gold correction implies a reference to a specific wedding, possibly the one involving the interlocutor. By contrast, the use of 'di', correct in the model's output, generalizes the meaning, resulting in a broader, more abstract question about marriage as a concept.

Examples from sentence(52) to sentence(55) display cases in which the model missed an incorrect structure within the sentence. In sentence (52), the learner uses the simple preposition 'di', omitting the required contracted form 'dell''. The model fails to flag this error. Sentence (53) is more ambiguous with regard to error classification. According to the MERLIN dataset, the token 'in' is marked as incorrect due to the expected presence of an article. However, the most common way of expressing this concept in Italian would use a different preposition entirely (i.e. 'a'). Although one could argue for the form 'andare al calcio' (*to go to the football*), suggested in the annotator's correction, which would make this an article-related error, the more idiomatic expression in Italian is 'andare a calcio' (*to go to football*). This suggests that the error is better classified as one involving prepositional choice rather than article omission. Some errors also involve the incorrect selection of articles, either due to phonetic constraints or incorrect gender or number agreement. For example, the use of 'gli' before a feminine plural noun like 'Alpi' is incorrect, as it requires the feminine plural article 'le'. These examples demonstrate that the model struggles not only with identifying whether an article is required, but also with selecting the correct form when one is needed.

Finally, the model has difficulty with article usage involving possessive adjectives, especially when referring to family members. For example, in sentence (55), the model accepts superfluous articles before singular family member references without detection. Similarly, in sentence (56), the model fails to detect missing articles before plural family names.

This reflects a lack of sensitivity to a well-established rule in Italian grammar: possessive adjectives generally require a definite article (e.g., 'la mia casa', 'i miei amici'), except when referring to singular family members (e.g., 'mia madre', 'mio fratello'), where the article is typically omitted.

Generative Model Evaluation on Article Errors

As previously anticipated in Table 5.2, article-related errors present a moderate challenge for Qwen: of the 172 generated outputs, only 20 (11.6%) are fully correct and 10 (5.9%) partially address the issue, while the majority (60.4%) of outputs leave the error unresolved.

(57) Parteciperò volentieri a tutti **i** due **eventi**.
(a) Parteciperò volentieri a tutti e due gli eventi.
(b) Parteciperò volentieri ai due eventi.
I will gladly participate to both the events.

(58) perché non parli un po' di **matrimonio** quando telefoni con me?
(a) perché non parli un po' di matrimonio quando parli con me
(b) perché non parli un po' di matrimonio quando mi telefoni
Why don't you talk a little about your wedding when you call me?

(59) **Nostre** insegnante e **bella** ragazza.
(a) Nostre insegnanti sono belle ragazze.
(b) Nostre insegnante è bella ragazza.
(c) Nostre insegnante è una bella ragazza.
Our teacher is a beautiful girl.

(60) La sera io posso andare {{al}} **un** concerto con **{{la mia}}**padre o andare in un ristorante con la mia famiglia
(a) La sera io posso andare a un concerto con la mia madre o andare in un ristorante con la mia famiglia.
(b) La sera io posso andare al concerto con la mia sorella o andare in un ristorante con la mia famiglia.
In the evening I can go to the concert with my dad or go to the restaurant with my family.

The first correction (a) of sentence (57) demonstrates a direct and accurate fix that makes the sentence fully grammatical. The second correction (b), while also producing a fully correct result, takes a more radical approach by removing the quantifier 'tutti' and restructuring the prepositional phrase, essentially simplifying the entire construction rather than making targeted adjustments to the original structure.

In some cases, however, the model does not address all the errors present in the sentence. Issues related to article usage may be bypassed, and the model instead provides only partial corrections. This is often due to the varied nature of the errors, which may involve different linguistic components and lead to shifts in meaning. This is the case for sentence (58). The original sentence contains a prepositional not so fluent use following the verb 'parli'. However, the sentence could still be interpreted meaningfully, though likely not in the way the learner intended. The use of 'del' in the gold correction implies a reference to a specific wedding, possibly the one involving the interlocutor. By contrast, the unchanged use of 'di' in the model's output generalizes the meaning, resulting in a broader, more abstract question about marriage as a concept. Although this semantic shift alters the scope of the utterance, from a specific event to a general idea, the resulting sentence remains grammatically valid.

Notably, some sentences, like sentence (59), elicit no usable corrections from the model. For instance, none of the five generated corrections successfully resolve all the central issues together: the incorrect possessive adjective ('nostre' instead of 'nostra'), the missing article before the noun phrase ('una bella ragazza', *a beautiful girl*), the lack of a definite article before the possessive ('la nostra insegnante', *our teacher*), and the incorrect verb conjugation 'è' (*she is*) instead of 'e' (*and*). While a few corrections make partial improvements, such as changing 'e' to 'è' or adding 'una', they either leave the possessive adjective unchanged or introduce agreement mismatches elsewhere (e.g., 'Nostre insegnanti sono belle' shifts to plural without justification, and still omits the article before the possessive). This case illustrates the model's inability to produce coherent or contextually appropriate edits when multiple interacting errors are present, despite their explicit annotation in the gold labels.

A subset of eight sentences in the dataset exhibiting both false positives and false negatives, offers a more complex picture of model behavior. These cases are valuable because they reveal scenarios in which the model partially recognizes issues but fails to provide complete or accurate corrections.

For example, in (48), the gold correction removes the unnecessary article ('il') to yield 'Quanto costa questo viaggio?' (*How much does this travel cost?*). Qwen offers multiple plausible corrections here, including one that drops the article ('Quanto costa questo viaggio?') and another that drops the demonstrative adjective ('Quanto costa il viaggio?', *How much does*

the travel cost?). Both forms are grammatically correct, illustrating that the model recognizes alternative strategies for resolving the error. However, the model flags 'il' as incorrect rather than 'questo', which is not aligned with the gold label but still acceptable in terms of outcome.

Another instructive example comes from sentence (60). The gold correction involves fixing both the article contraction ('al un' → 'al', as in the minimal target hypothesis or → 'a un', as in extended target hypothesis) and the gender disagreement in the possessive construction ('la mia padre' → 'mio padre'). The gold correction involves two distinct modifications: first, the resolution of an article contraction error, where the ungrammatical sequence 'al un' is corrected to either 'al' (*to the*, as in the minimal target hypothesis) or 'a un' (*to a*, as in extended target hypothesis). Second, it addresses a gender mismatch in the possessive construction, replacing the incorrect 'la mia padre' with the grammatically appropriate 'mio padre'. The model, however, avoids correcting the gendered components directly. Instead, it substitutes the noun 'padre' with 'madre' (*mother*) or 'sorella' (*sister*), sidestepping the agreement issue by introducing a feminine referent rather than aligning the article and possessive with the masculine noun. This shift maintains grammaticality but alters the semantic content. Moreover, it consistently fails to remove the unnecessary article before the possessive adjective ('la mia madre'). As such, the corrections provide partial grammatical repairs but do not fully address the target errors as annotated in the gold labels.

5.3.2 Grammar_Agreement

Agreement-related errors are relatively infrequent, with only 15 instances observed across 9 sentences in the dataset. These error type typically involves verb and noun mismatches, such as subject-verb agreement in person, number, or tense, as well as auxiliary selection and participle agreement, particularly in compound tenses and passive constructions. Of the 15 instances, 8 occur in sentences containing only false negatives, as previously discussed in Table 4.13, and 1 appears in a sentence that includes both false positives and false negatives. As shown in Table 5.6, among these 9 sentences, 3 are grammatically correct despite the MultiGED labels marking some tokens as incorrect, while the remaining 6 are cases in which the model misclassifies the tokens.

(61) E quando **abbiamo** deciso di sposarvi?
And when have you decided to get married?

(62) Invece, l'inglese, che lo **aveva usato** prima moltissimo quando **lavorava**, solo l'uso quando stiamo con gli amici stranieri che non parlano italiano.
However, English that I used a lot before when I worked, I use it only when we are with foreigner friends that do not speak Italian.

(63) Probabilmente **arrivò** il 18 novembre e resterò fino al 22 novembre.
Probably I will arrive on the 18th of November and will stay there until the 22nd November

(64) Ma tutto era al contrario come {{era}} **scritta** nel Suo annuncio.
But everything was the opposite of how it was written in Your announcement.

Sentence (61) is a clear example of a construction that, although awkward in meaning, can be considered grammatically correct. In this case, the model fails to detect the mismatch between 'abbiamo' (first-person plural) and 'sposarvi' (second-person plural object). While the sentence is structurally well-formed, its interpretation is ambiguous and pragmatically odd. It could be interpreted either as a reflexive infinitive (i.e., 'sposarvi', *you get married*)), implying that the subject and object coincide (a second-person plural action), or as a transitive verb where 'vi' is the direct object (i.e., *we decided to marry you*), suggesting a first-person plural subject acting upon a second-person plural object. The latter interpretation would imply an agentive role, such as that of a wedding officiant. This ambiguity, combined with the lack of an explicit subject, likely contributes to the model's failure to flag either an agreement or valency error. These

cases underscore the model's difficulty in detecting such mismatches when sentence structure is complex or when interpretation relies on implicit discourse elements. Additional examples of this phenomenon appear in the analyzed section.

Sentences reveal a consistent pattern in the model's wrong predictions: the model struggles when the error occurs inside a subordinate or relative clause (62), or when the subject of a verb is implicit (63).

In sentence (62), 'Invece, l'inglese, che lo aveva usato prima...', the model fails to identify that the relative clause 'che lo aveva usato prima' contains a subject–verb agreement error. The verb 'aveva' is in the third person singular, yet the intended subject is the speaker, consistent with the main clause ('uso'), and thus the correct form would be 'ho usato'. This demonstrates a lack of contextual and syntactic awareness, as the model does not propagate subject information across clauses. Additionally, the sentence contains several other problems, such as incorrect valency, punctuation, and word order, which go completely undetected, highlighting the model's difficulty with multi-layered grammatical constructions.

Additionally, in sentence (63) the model fails to detect the possible agreement mismatch between the past tense of the verb 'arrivò' (*he/she arrived*) and the implied subject. In this case, there are two plausible sources of error: either 'arrivò', the third person singular of the past tense, is incorrect and should be replaced with 'arriverò' (*I will arrive*), which is the first person singular of the future tense, or the second verb 'resterò' (*I will stay*) might be at fault and could be corrected with 'restò' (*he/she stayed*). Since the subject is not explicitly stated and the two verbs are coordinated by the conjunction 'e', resolving the intended subject requires discourse-level inference. This ambiguity may explain why the model does not flag any error, despite the gold annotation marking 'arrivò' as incorrect.

The only case, sentence (64), with both false positives and negatives offers an instructive example of how the model struggles with interacting grammatical dependencies. The subject 'tutto' (*everything*) is masculine singular and correctly agrees with the verb 'era' (*was*). However, the model misclassifies 'era' as incorrect while failing to flag 'scritta' (*written*) as an error, even though the participle should agree with the masculine subject and thus be 'scritto'. Additionally, the model does not flag the preposition 'al', a contraction of 'a + il', which may be contributing to the confusion. The correct version, as annotated in the gold labels, replaces 'al' with 'il' and 'scritta' with 'scritto'. The combination of nominal agreement, participle morphology, and prepositional contraction in a single sentence may have overloaded the model's ability to resolve grammatical dependencies accurately.

Generative Model Evaluation on Agreement Errors

As noticeable in Table 5.2, none of the 43 generated outputs in this batch are fully correct in terms of matching the gold corrections. However, in several cases, the model produces grammatically well-formed outputs that are semantically plausible but diverge from the target meaning:

(65) Mi immaginerei un corso in un gruppo nuomente piccolo - 4 persone al massimo - se questo non è possibile **prenderai** dei lezioni individuali.
(a) Mi immaginerei un corso in un gruppo nuovamente piccolo - 4 persone al massimo - se questo non è possibile prenderai delle lezioni individuali.
(b) Mi immaginerei un corso in un gruppo nuovamente piccolo - 4 persone al massimo - se questo non è possibile prenderesti delle lezioni individuali.
I would imagine a course in a numerically small group — 4 people at most — if that is not possible, I would take individual lessons.

(66) Se avete **altri cosi** che vi **faccio** piacere potete dirmi.
(a) Se avete cose altre che vi piacciono potete dirmi.
(b) Se avete altri cosi che vi piace potete dirmi.
If you have other things (f.) that you like you can tell me.

(67) Secondo me i tuoi amici perdonano e non ti chiedono più perché non ti **faccio** più sentire.
(a) Secondo me i tuoi amici perdonano e non ti chiedono più il motivo per cui non ti faccio più sentire.
(b) Secondo me i tuoi amici perdonano e non ti chiedono più il motivo per cui non ti faccio più sentirti.
In my opinion your friends forgive you and don't ask anymore why you are not talking to them anymore.

Corrections (a) and (b) both improve on the original sentence (65) by correcting lexical and morphological errors, such as replacing the non-existent adverb 'nuomente' with 'nuovamente' (*again*), and fixing the article-noun mismatch ('dei lezioni' con 'delle lezioni'). However, both outputs maintain or introduce a person mismatch in the second clause ('prenderai', *you will take*/'prenderesti', *you could take* instead of 'prenderei', *I would take*), shifting the subject from first to second person. While these versions are grammatically well-formed and fluent, they diverge from the intended meaning and therefore cannot be considered fully correct under the current evaluation scheme. Nonetheless, they illustrate the model's ability to generate acceptable alternatives, even when semantic fidelity is not fully preserved.

Among partially corrected instances, sentence (66) illustrates different types of incomplete fixes. Correction (a) represents a case where the model successfully addresses the verb agreement by changing the verb to 'piacciono', but simultaneously introduces new errors: it creates a word order problem ('cose altre' instead of the correct 'altre cose'). Correction (b) replaces the problematic construction 'vi faccio piacere' (*I please you*) with the verb 'piacere' (*to like*), but uses the wrong agreement form 'piace' (third person singular) instead of the required 'piacciono' (third person plural) to match the plural subject *altri cosi*.

In sentence (67), the model does not correct anything, it attempts to clarify the final clause using structures like 'il motivo per cui' (*the reason why*), which was not wrong in the first place, but leaves the core valency error unresolved, incorrectly retaining 'ti faccio' (*I let you*) instead of the reflexive 'ti fai' (*you do*).

In summary, the model fails to produce any fully correct outputs in this sample. While partial corrections are relatively common, they are typically shallow or inconsistent, often correcting surface-level morphology or word choice while ignoring deeper syntactic, agreement, or valency issues. This indicates that while the model is sensitive to some forms of local grammatical error, it lacks a reliable mechanism for coordinating multiple edits or reasoning over sentence structure and meaning.

5.3.3 Grammar_Verb

Across the dataset, verb-related errors represent one of the most frequently missed categories, with a total of 32 incorrect instances (false negatives) out of 37 annotated cases. These errors span 20 unique sentences where the model either fails to detect any verb errors or produces only partially correct outputs. Additionally, there are 5 sentences where both false positives and false negatives are present, indicating inconsistent behavior. Common error patterns include tense mismatches, especially between past and present forms (e.g., *ho scritto* → *scrivo*; *è stato* → *era*), incorrect verb inflections (*potrai* → *puoi*; *faresti* → *fai*), and auxiliary selection errors (*ha potuto* → *è potuta*).

In particular, this error type refers to cases in which the tense of the verb does not align with the intended context or temporal frame of the sentence. It is reasonable to expect the model to struggle here, especially when determining whether the tense of the main verb is contextually appropriate. The gold annotations often rely on discourse-level information, such as the speaker's temporal intent or narrative context, information that is not available to a model operating at the sentence level. Therefore, certain constructions that require the past tense in full context may still appear grammatically correct when isolated, potentially leading to false negatives. However, this contextual justification does not extend to subordinate clauses, where tense is typically constrained by that of the main clause. As previously seen (62), the

model exhibits particular difficulty handling tense consistency across subordinate structures and fails to model dependencies between main and embedded clauses.

Within the 25 analyzed sentences (see Table 5.6), 6 are grammatically correct, meaning the model cannot be held accountable for the incorrect predictions, while the remaining 19 contain genuine errors.

(68) Invece **potrai** dirmi se avete fatto una lista di nozze o qual regalo vi farà piacere.
Instead you can to tell me if you have made a wedding list or which gift you will like.

(69) Io lo sapevo, che tu ce la **fai** senza problemi.
I knew that you could make it without any problem.

(70) Ho cominciato a studiare l'inglese molto presto, quando andavo all'asilo e **continuavo** anche dopo, fino a quando **avevo** la possibilità di andare all'estero.
I have started studying English very early, when I was in kindergarden, and I continued even later, until when I had the chance of going abroad.

Sentence (68) is annotated as containing a verb error, suggesting that 'potrai' (*you will be able to tell me*), the future, should be corrected to 'puoi' (*you can*), the present. While 'puoi' may be more contextually appropriate, 'potrai dirmi' is a fully grammatical future-tense construction. The model does not flag this as an error, and arguably, this abstention is reasonable. However, without access to the surrounding context, both versions are grammatically acceptable.

Sentence (69) and sentence(70) contain genuine misclassifications from the model. For instance, in the example above (69) after the main verb 'sapevo' (*I knew*, in the imperfect), the model should have detected that the prepositional verb 'ce la fai' (*you can make it*, in the present) is incorrect, as it violates the rules of *consecutio temporum*. This occurs also in other circumstances with the use of the subjunctive in subordinate clauses. Sentence (70) highlights a recurring issue in the dataset, and one often seen in Italian learner corpora, which is the alternation between *imperfetto* and *passato prossimo*[6]. The model also appears to struggle in determining when one tense is preferable to the other. The learner (70) alternates between the *passato prossimo* (e.g., 'ho cominciato') and the *imperfetto* (e.g., 'andavo', 'continuavo', 'avevo') without fully mastering their aspectual distinctions. While each individual clause may appear grammatically correct in isolation, the overall temporal progression is confusing. The model fails to flag this incoherence, likely because all the tense forms are individually correct. However, appropriate coordination of verb aspects is crucial for maintaining clear temporal structure in Italian. Misuse or inconsistent combination of *passato prossimo* and *imperfetto* remains a known challenge among Italian learners, and this again underscores the model's limitations in evaluating inter-clausal aspectual harmony and narrative sequencing, especially when all surface forms are technically grammatical.

The five sentences that contain both false positives and false negatives, in this case, do not provide any additional insight on the model's behavior. Issues arises always in case of subordinate clauses, and in the alternation of *passato prossimo* and *imperfetto*.

Generative Model Evaluation on Verb Tense Errors

For this error type, the model produces only 12 fully correct sentences and 11 partially correct ones out of 117 generated outputs (see Table 5.2). The remaining 94 outputs either fail to address the targeted error, focusing instead on superficial features such as spelling, or modify irrelevant aspects of the sentence that do not resolve the underlying issue.

(71) In augosto faceva molto caldo, ma l'aria condizionata non **ha funzionato** nella nostra camera.
(a) In agosto faceva molto caldo, ma l'aria condizionata non funzionava nella nostra

[6]In English, the Italian *imperfetto* often corresponds to the past continuous (e.g., 'was going') or the expression 'used to', while the *passato prossimo* typically maps to the simple past (e.g., 'went') or present perfect (e.g., 'have gone'), depending on context.

camera.
In August it was very warm, but the air conditioning was not working in our room.

(72) Puoi dirmi dove **hai lavorato** adesso?
(a) Puoi dirmi dove lavori adesso?
(b) Puoi dirmi dove hai lavorato ultimamente?
Can you tell me where have you worked recently?

(73) *Vi ringrazio per l'invito e mi **piacere** venire.*
(a) Vi ringrazio per l'invito e mi piacerebbe.
(b) Vi ringrazio per l'invito e mi piacerebbe di venire.
(c) Vi ringrazio per l'invito e sarei onorato di venire.
I thank you for the invitation and I will like to come.

In relation to the wrong use of *imperfetto* and *passato prossimo*, sentence (71) shows how the model finds multiple corrections that are perfectly aligned with the gold labels. In this specific scenario, Qwen managed to address more of the errors than XLM-RoBERTa.

Among the fully corrected sentences, the model also managed to provide multiple possible corrections by minimally modifying the tokens, without altering the overall sentence structure. This is the case of sentence (72), where Qwen acts in two different ways: either by modifying the verb tense or by adjusting the related adverb at the end of the sentence.

For sentence (73), Qwen produces corrections that could be plausible, even though not necessarily aligned with the gold corrections. Three of the five generated corrections are provided; the first and the last are completely correct, even though the first seems to be more coherent with the original structure of the sentence, while the second introduces more modification. In particular, the first one does not match the gold correction provided in MERLIN, which makes use of the future 'piacerà' (*I will like*). Probably, based on the context (which we are not aware of), the future tense is more correct than the conditional. However, the future is not very frequent with this specific verb. To express politeness and kindness, it is more appropriate to use the conditional, which is what Qwen offers as a possible correction: 'piacerebbe' (*I would like*). The second correction introduces an error by adding the preposition *di* before the infinitive.

Overall, while the model shows promising capacity to generate plausible corrections for verb-related issues, it remains inconsistent. It often fails to coordinate verb tenses across main and subordinate clauses, and struggles with subjunctive mood and aspectual consistency. The same limitations are evident in sentences that contain both false positives and false negatives.

5.3.4 Grammar_Valency

Valency errors in the dataset reveal systematic issues with the model's understanding of argument structure, particularly in verb complementation, reflexivity, and auxiliary selection. The following examples focus on sentences that contain only false negatives or fully correct predictions. Five additional cases contain both false positives and false negatives, but do not provide any additional insight into how the model handles valency errors. The incorrectly flagged tokens in these cases are unrelated to the underlying valency issues, and therefore do not contribute meaningful evidence toward understanding model behavior. Therefore, only the sentences that contain false negatives and correct predictions are further analysed.

Out of the 17 sentences analyzed (see Table 5.6), 2 show the model suggesting a different but plausible correction, 5 are grammatical, and in 10 the model makes incorrect predictions.

A representative example has already been mentioned in (67), where the model fails to revise the core valency error in 'perdonano' (*they forgive*), which requires two arguments: a nominal subject, 'i tuoi amici' (*your friends*), and a direct object, 'ti' (*you*), which is missing here. Further examples are presented below:

(74) Sa-unreadable-ete essatamente che voi non {{avete}} ancora. *Unreadable exactly that you do not have yet.*

(75) Certo io posso **aiutare**.
Sure, I can help you.

(76) **Non** sto bene è una settimana orribile.
I am not fine, it's a horrible week.

(77) Per questo grande passo vi auguro profondamente**.**
For this big step I wish you deeply the best.

Sentence (74) represents the unique case of plausible alternative prediction. The sentence is not fully readable, which prevents a complete understanding of its meaning. However, the relative clause contains the verb 'avete' (*to have*), which is not followed by a direct object. The MultiGED annotations do not account for this and label the token as correct. In contrast, the model marks it as incorrect. This incorrect label can nonetheless be plausibly mapped to a valency error in the original MERLIN dataset, which suggests that the model's prediction is not entirely unjustified.

Sentences (75) and (76) can be considered grammatical. In sentence (75), for instance, the verb 'aiutare' (*to help*) appears without an explicit direct object. However, this does not result in ungrammaticality. In this context, the meaning is fully acceptable and commonly used in Italian, especially in informal conversation. The sentence does not seem to require further specification of who or what is being helped. Sentence (76) is fully grammatical except for a missing comma between 'bene' (*fine*) and 'è' (*it is*). The token 'Non' is flagged as incorrect and possibly related to a valency error, according to the MERLIN dataset; however, without access to the broader context, this cannot be confirmed. In isolation, the sentence appears grammatical.

Sentence (77) is an example of a wrong prediction. It contains the predicate 'auguro', which lacks the third necessary argument: the direct object. The gold label highlights this error by marking the final period as incorrect, signaling that essential information is missing before the sentence concludes. The model, however, fails to make such a suggestion and labels all tokens as correct. This type of issue appears in a total of five cases. In two other cases, the model fails to flag the incorrect presence of an extra, redundant argument, as in example (62).

These observation suggests that some instances labeled as valency errors in the dataset may stem from annotation inconsistencies or overly broad mappings from the original MERLIN scheme to the MultiGED label taxonomy.

Generative Model Evaluation on Valency Errors

As for the rest of the generated corrections, Qwen produces a mix of plausible, partially correct, and clearly incorrect outputs for valency-related issues (see Table 5.2):

(78) **Conosco un** {{veramente}} buonissimo!
(a) Conosco un buonissimo!
(b) Conosco un veramente buono!
(c) Conosco un buonissimo restaurant!
I know one really good!

(79) Gli occhi parlano molto e **si** danno informazioni importanti per giudicare una persona.
(a) Gli occhi parlano molto e danno informazioni importanti per giudicare una persona.
(b) Gli occhi parlano molto e forniscono informazioni importanti per giudicare una persona.
Eyes speak loud and convey important information for judging a person.

(80) Certo io posso **aiutare**.
(a) Certo io posso aiutarti.
(b) Certo io posso aiutare qualcuno.
(c) Certo posso aiutare.

Sure, I can help.

(81) **{{Prego}}** di rispondere sulla mia lettera il più presto possibile.
(a) Prego di rispondere alla mia lettera il più presto possibile.
(b) Prego di rispondere alla mia lettera al più presto.
(c) Prego di rispondere alla mia lettera con la massima rapidità.
I kindly ask you to respond to my letter as soon as possible.

While several candidates closely approximate the intended meaning and improve on the original sentence, others fail to address the core valency violation or introduce unrelated errors, this occurs especially in relation to sentences that contain both false positives and false negatives, as visible in sentence (78). Especially in correction (c), it is noticeable that the model recognizes the valency issue associated with the verb *conosco* ('I know') and attempts to resolve it by adding a direct object: 'restaurant'. However, it inserts the object in English rather than Italian, indicating a partial but flawed understanding of the syntactic requirement.

By contrast, Qwen performs remarkably well on sentence (79), where XLM-RoBERTa only assigned correct (*c*) labels and failed to detect the redundant reflexive pronoun. Qwen correctly identifies the valency violation and removes the extra argument. Additional variants introduce lexical diversity by replacing 'danno' with 'forniscono', which remains semantically appropriate. Other corrections in this set replace the indefinite article with a definite one, slightly altering the sentence's meaning but without introducing grammatical errors.

In another example where XLM-RoBERTa assigned all tokens a correct label, Qwen partially addresses the valency issue in (80). The model introduces a direct object either as a clitic pronoun or an indefinite noun. Even the variant without any object cannot be considered entirely incorrect. In fact, without additional context, the original sentence might not be interpreted as erroneous at all.

In the case of (81), Qwen focuses on the incorrect preposition 'sulla' and replaces it with more appropriate alternatives. Unlike XLM-RoBERTa, which flags the lack of a direct object in 'prego', Qwen does not explicitly add a missing argument in any of its variants. All generated options correctly handle the problematic preposition and offer idiomatic alternatives for the adverbial phrase. However, none of them address the underlying valency error associated with the verb 'prego', which in this context requires an explicit object (e.g., 'ti prego', *I ask you*). Only correction (a) retains almost the entire structure of the original sentence, but the valency issue remains unresolved in all outputs.

Overall, Qwen is capable of generating high-quality corrections, but with significant variability. Some outputs reflect minimal yet sufficient edits, while others overcorrect or shift meaning unnecessarily. Valency errors prove especially challenging, as they often require adjusting clitics, argument realizations, and verb morphology simultaneously.

5.3.5 Grammar_Word-Order

Word order errors are among the most frequent error types in the dataset, occurring in 17 sentences. These errors are often caused by unnecessary or misordered constituents that disrupt the canonical sentence structure. While they do not always result in ungrammaticality, they frequently reduce fluency or pragmatic appropriateness. Consequently, the model may diverge from the annotator in its error classification (see Table 5.6), either by attributing the error to a different token (3 cases), producing false negatives when the sentence remains grammatical (6 cases), or failing to flag the error altogether (8 cases). This section evaluates the model's sensitivity to such disruptions and its overall behavior in handling them.

(82) Benché io abbia tanto da fare al lavoro, non vorrei evitare di essere parte della **festa {{grande}}**.
Although I have a lot to do at work, I would not want to avoid being part of the big party.

(83) In attesa di una risposta **buona** porgo i miei cordiali saluti.
I look forward to your reply. Kind regards.

(84) Sono euphorica che volete stare insieme **sempre** e che i genitori annunciate il matrimonio **finalmente**!
I am thrilled that you want to stay together and that your parents are finally announcing the wedding!

(85) Vogliamo **qui** andare a cena sabato sera?
Do we want to go to dinner here on Saturday evening?

(86) Ma non **c'** è così tutto che abbiamo pensato.
But everything did not turn out as we expected.

Example (82) is one of three cases in which the model appears to propose a different correction, or perhaps a partial suggestion, by labeling only the adjective 'grande' (*big*) as incorrect. Although 'festa grande' (*big party*) is syntactically acceptable, it violates idiomatic preferences in Italian word order. Adjectives of size and importance, such as 'grande', tend to precede the noun when used figuratively or evaluatively (e.g., 'una grande festa', *a big party*), while post-nominal placement usually indicates a literal, restrictive interpretation. The phrase 'festa grande' therefore sounds non-native or overly literal. In this case, the model correctly flags 'grande' as erroneous, matching the gold annotation. This represents a rare instance in which the model shows sensitivity to stylistic infelicity at the noun phrase level, beyond grammatical correctness.

Examples (83) and (84) are grammatically correct sentences, even if they are somewhat of a stretch. Despite being grammatical, sentence (83), which is an instance of formal writing, appears odd because word order within the noun phrase can influence tone, register, and stylistic naturalness. While the phrase 'risposta buona' (*good answer*) is grammatically correct, it is pragmatically marked and sounds awkward in formal written Italian. In such contexts, attributive adjectives like 'buona' (*good*) are typically placed before the noun (i.e., 'una buona risposta', *a good answer*), or omitted altogether when redundant or formulaic. The gold correction removes the adjective entirely, restoring a more conventional and stylistically appropriate closing. The model, however, leaves the input unchanged, failing to account for the pragmatic dimension of adjective placement.

In sentence (84), issues with topicalized or repeated elements are also overlooked. The sentence contains multiple problems beyond surface-level errors, most of which pertain to word order and pragmatic naturalness. While 'euphorica' is clearly a spelling error for 'euforica', the main source of disfluency lies in the placement of the adverb 'sempre' (*always*), the verb 'annunciate' (*you announce*), and the adverb 'finalmente' (*finally*). The sequence of subordinate clauses is grammatically recoverable but pragmatically awkward, particularly due to the post-verbal position of 'finalmente', which produces a marked or emotionally flat reading. In idiomatic Italian, 'finalmente' would more naturally precede the verb ('finalmente annunciate il matrimonio') or appear earlier in the clause structure. This example illustrates the model's limited sensitivity to subtle disruptions in information structure, especially when lexical and morphological cues remain plausible. Adverbs that contribute to discourse coherence or emotional emphasis, such as 'sempre' and 'finalmente', pose a particular challenge: their misplacement does not lead to ungrammaticality but significantly reduces naturalness and clarity.

Examples (85) and (86) are cases in which the sentences are more noticeably ungrammatical and the model fails to identify the issue at all. For instance, sentence (85) shows a redundant adverb inserted in a question, which is syntactically invalid and pragmatically misplaced. The gold correction removes the adverb 'qui', as it disrupts the idiomatic flow of the sentence. The model, however, fails to flag it, suggesting low sensitivity to such pragmatic word order issues. Sentence (86) demonstrates how the model ignores deeper structural issues involving expletives and non-canonical clause ordering. The phrase 'così tutto che' (*everything as such that*) is unidiomatic and suggests confusion in clause-level structure. The gold correction rewrites this segment with a more canonical form 'come ci aspettavamo' (*as we were expecting*). The model,

however, does not flag the problem and fails to classify the relevant tokens as incorrect.

Generative Model Evaluation on Word Order Errors

In Table 5.2, one can see that this error class is where Qwen performs the worst. Out of 85 generated sentences, only 8 are fully correct, and none are partially correct. The remaining 73 generations fail to address the word order issue, instead modifying irrelevant or already correct elements, or leaving the sentence unchanged altogether.

In a few cases, Qwen shows partial sensitivity to awkward sequences, particularly in modifying or omitting adjectives. One example, in sentence (87) involves the phrase 'il giorno grande' (*the big day*):

(87) Sono sicuro che loro aspettano molto "il **giorno** {{grande}}".
(a) Sono sicuro che loro aspettano molto il giorno grande.
(b) Sono sicuro di loro aspettativa di un giorno grande molto.
I'm sure they're really looking forward to "the big day".

All of the generated corrections fail to fix the original word order error. The phrase 'il giorno grande' is not idiomatic in Italian; the expected form would be 'il grande giorno'. None of the outputs reflect this, and several introduce additional stylistic or syntactic issues, such as overly complex and nonsensical nominal constructions like in (b).

In more formal contexts (83), however, Qwen occasionally applies more appropriate reordering, by inverting the order of 'risposta buona' to 'buona risposta'.

In this case, several variants correctly invert the adjective-noun order or delete the adjective entirely, aligning more closely with idiomatic Italian usage. This suggests that Qwen performs better when dealing with frequent collocations or formulaic expressions, especially in high-register text.

Moreover, when more substantial reordering is needed, especially across clause boundaries, Qwen consistently fails to apply the required transformations. For example in sentence (86), the malformed sequence 'così tutto che' (*so all that*) remains untouched in all generated variants, or is replaced with similarly implausible alternatives such as 'così tanto che' (*so much that*), 'così completo che' (*so complete that*). This indicates difficulty with clause-level restructuring and suggests the model prefers local lexical substitution over global reordering.

A similar pattern appears in sentences where redundant adverbs should be removed as for sentence (85), on which Qwen either retains the misplaced adverb 'qui' (*here*) or inserts it in equally wrong positions (e.g., 'qui andare a sera'). This further supports the observation that Qwen struggles with pragmatic word order correction.

Overall, Qwen shows limited capacity to resolve word order errors when corrections involve non-local dependencies or discourse-level restructuring. Most successful outputs operate at the phrase level and reflect collocational familiarity rather than deeper syntactic understanding.

5.3.6 False Positives – Italian

The subset of Italian sentences containing only false positives and correct predictions includes 42 instances, which constitutes a small proportion of the total number of sentences in the processed development set. Within this group, several recurrent patterns emerge (see Table 5.7): in 10 cases, the model's prediction differs from the gold labels in a way that suggests an alternative plausible correction; in 10 other cases, the model's prediction reveals misannotations in the MultiGED dataset; and in 22 cases, the model makes a wrong prediction. This finding suggests that relying exclusively on confusion matrices may underestimate the model's true effectiveness.

Error Type	Sentences	Model Correct: Plausible Correction	Model Correct: Mislabeled in MultiGED	Model Wrong
False Positives	42	10 (23.8%)	10 (23.8%)	22 (52.4%)

Table 5.7: Distribution of XLM-RoBERTa false positive predictions, grouped by evaluation category (MERLIN): plausible correction, mislabeled in the MultiGED annotations, or incorrect prediction.

(88) Tanto non **{{bisognò}}** {{di}} prenotare un alloggio.
Anyway, there is no need to book accommodation.

(89) Secondo me l' amicizia virtuale sia possibile solo in {{alcune}} **{{parte}}**.
In my opinion, virtual friendship is possible only in a few parts.

(90) Ciao {{Caro}}, come stai?
Hello Dear, how are you?

(91) Ciao Michele grazie per la tua {{.....}} .
Hi Michele, thank you for your...

(92) Non abbiamo dei gravi problemi come la {{famina}}.
We do not have serious problems such as starvation.

(93) Io {{viaggo}} IN TRENO, IO SONO ALLE 19 A CASA TUA.
I travel by train; I will be at your house at 7 p.m.

(94) Vorrei presentarmi {{in}} breve e darVi alcuni informazioni sulla mia persona.
I would like to introduce myself briefly and give you some information about myself.

One case of disagreement between model and gold labels which appears to stem from differing correction strategies is example (88). The token 'bisognò' (*needed*) is marked as incorrect in the gold, but the model also flags the following preposition 'di' (*of*). This divergence may arise from different plausible corrections. One approach, the one to which the annotator's correction is referring to, would replace 'bisognò' with 'bisogno' (*the need*) and add the auxiliary 'c'è' (*there is*), preserving the preposition 'di'. An alternative correction, changing 'bisognò' to 'bisogna' (impersonal present), renders the use of 'di' ungrammatical. In this light, the model's decision is defensible, depending on which correction path is assumed.

Another case of plausible classification is visible in sentence (89), in which the model flags both tokens involved in agreement errors, while the gold annotation marks only the second one. This is especially relevant when the correction would require changing both elements, which could lead to a more distant version from the learner's original. In this case, the model flags both the determiner and noun, while the gold annotation may mark only the noun.

Examples from (90) to (93) are cases where the MultiGED dataset is missing out some errors, which are wrongly classified as correct. One frequent pattern involves the model correctly identifying capitalisation errors, as in sentence (90), that were not annotated as such in the gold labels. Unless 'Caro' is intended as a proper name, the capitalisation is incorrect in standard Italian usage. The model flags this inconsistency, while the gold annotation does not. Punctuation inconsistencies are also often more reliably caught by the model, like in example (91), where the model identifies the unusual punctuation sequence as incorrect, while the gold annotation overlooks it. Other instances involve inexistent words (see sentence (92)) and clear spelling errors (see sentence (93)) that are missed by the gold labels. In sentence (92), the word 'famina' does not exist in Italian; the correct form is 'fame' (*hunger*), which goes uncorrected in the first correction layer of the MERLIN dataset. The model, however, aligned with the second correction layer, correctly flags the, whereas the gold annotation, which as mentioned derives from the first layer, fails to do so. In sentence (93), the misspelling of the verb 'viaggio' (*I travel*) is unmarked in the gold labels, but is accurately detected by the model.

Some false positives are genuine model errors, as in sentence (94). The preposition 'in' is part of the idiomatic expression 'in breve' (*briefly*), and thus should not be flagged. This type of error was observed in 12 cases, indicating that the model occasionally misidentifies fixed expressions.

Other examples are ambiguous due to their abstract phrasing and limited context (see sentence (95)) or due to annotation incoscistencies that may stem from preprocessing steps designed to anonymise named entities (see sentence (96)). For example:

(95) Forse ti {{piaci}}?
Maybe you like it?

(96) So che potrebbe essere difficile trovare un posto per me, ma dopo il 29 agosto comincerò studiare {{nei}} Paese Y.
I know it could be difficult to find a place for myself, but after August 29 I will start studying in Country Y.

While sentence (95) is technically grammatical, its meaning is pragmatically odd and potentially confusing (*Maybe you like yourself?*), which may explain the model's decision to flag it.

In sentence (96), the substitution of place names with labels such as 'Paese Y' (*Country Y*) sometimes results in agreement mismatches. The plural preposition 'nei' (*in the*) requires a plural noun (e.g., 'Paesi Bassi', *Netherlands*). When followed by a singular substitute like 'Paese Y', the agreement becomes invalid. A similar issue arises with anonymised monument names.

These observations collectively suggest that false positives are not merely noise but often reflect valid linguistic judgments by the model. This finding underscores the importance of qualitative analysis in evaluating model performance and highlights the need for careful annotation review, especially when models appear to outperform gold labels.

Generative Model Evaluation on False Positives

As shown in Table 5.3, Qwen achieves its best performance on sentences containing false positives: out of 200 generated outputs, 51 (25.5%) are fully correct and 21 (10.5%) are partially correct, amounting to 36% of outputs.

The model handles several surface-level phenomena, such as spelling and capitalisation, remarkably well. As seen in Example (93), spelling errors are consistently and accurately corrected, even when the gold labels fail to detect them. Likewise, in cases involving incorrect capitalisation, such as in Example (90), Qwen systematically lowercases words that are improperly capitalised in the middle of a sentence. This behavior is consistent across all generated corrections, except when the capitalised word appears at the beginning of a letter, where it is correctly retained.

(97) Caro Daniele, **{{Mi}}** dispiace {{ma}} **{{e}}** una settimana orribile.
(a) Caro Daniele, Mi dispiace ma è una settimana orribile.
Dear Daniele, I am sorry, but it is a horrible week.

In more semantic cases, such as the invented word 'famina' in Example (92), Qwen produces varied yet appropriate corrections. While two of the five generated outputs retain the invalid form, the other three use the correct noun 'fame' twice and the less frequent but contextually valid synonym 'carestia' once. This suggests the model can interpret the sentence meaning and retrieve plausible lexical alternatives. However, in idiomatic expressions like 'ragazza alla pari', the model struggles. Instead of correcting the misspelling 'pare' to 'pari', it replaces the term with broader alternatives such as 'babysitter', or even switches to a French borrowing like 'au pair', showing some lexical flexibility but missing the specific idiom.

In agreement errors where MultiGED flags only one token, RoBERTa typically flags both tokens, Qwen only modifies the syntactic head during correction. For instance, in Example (89),

the model corrects the noun 'parte' (*part*) to its plural form 'parti' (*parts*), but leaves the determiner 'alcune' (*some*) unchanged. This suggests a minimalist correction strategy, which aligns with a preference for low-edit distance outputs but may diverge from exhaustive grammatical alignment.

Example (88) illustrates a case where Qwen diverges from the correction strategy implicit in the gold labels. While the annotation suggests inserting the predicate 'c'è' (*there is*) and replacing 'bisognò' with 'bisogno', none of Qwen's outputs follow this path. Instead, the model produces a more economical correction by replacing 'bisognò' with the impersonal present 'bisogna' and omitting the preposition 'di', reflecting a valid and arguably more direct repair. This demonstrates that the gold correction is not always the most natural or straightforward option from the model's perspective.

In cases where the original sentence is already grammatical, such as Example (94), where the idiomatic phrase 'in breve' (*shortly*) is misidentified by RoBERTa as erroneous, Qwen tends to preserve the sentence faithfully. When it does propose changes, they are usually paraphrastic, maintaining the core meaning but offering alternative formulations. This suggests that the model is relatively conservative in overcorrecting idiomatic content.

In contrast, ambiguous yet grammatically correct constructions, such as Example (95), present a challenge. The sentence 'Forse ti piaci?' (*Maybe you like yourself?*) is syntactically valid but semantically marked. Qwen's outputs tend to overgenerate speculative or nonsensical alternatives, failing to preserve the subtlety of the original construction. This reflects the difficulty of handling pragmatically unusual but grammatical input.

Finally, the case of anonymised entities, such as in Example (96), reveals a unique model behavior. Faced with artificial placeholders like 'Paese Y', Qwen produces multiple variations of the preceding preposition, covering a range of agreement scenarios (e.g., 'nel', 'nei', 'nello', 'in') that could potentially match the underlying country name. This shows the model's attempt to phonologically and syntactically harmonise unknown elements, an emergent adaptation to artefacts in anonymised data.

Overall, these examples reinforce the notion that Qwen's false positives often stem from meaningful grammatical or stylistic decisions, rather than misclassification. In many cases, the model's predictions offer valuable insights into annotation inconsistencies, idiomatic usage, and error correction strategies, further supporting the importance of qualitative analysis in evaluation.

Chapter 6

Discussion

This chapter offers a critical interpretation of the results presented in the previous chapters, linking them to broader trends in grammatical error detection and exploring their implications in the context of the MultiGED-2023 shared task. In addition to discussing key findings and model behavior, the chapter reflects on the limitations of this study and suggests directions for future research.

6.1 Overview of Findings

The experimental results demonstrate that while multilingual transformer models such as XLM-RoBERTa exhibit robust performance across languages, they do not consistently outperform strong monolingual baselines. However, it is important to acknowledge that the observed superiority of monolingual models in this study may also be partially attributed to differences in model size. While smaller monolingual models failed to reach competitive performance, their larger counterparts, bert-base-italian-xxl-cased for Italian and bert-large-cased for English, achieved notably strong results.

For both English and Italian, this thesis revealed a particularly strong performance of the models in surface-level patterns, such as orthographic errors, preposition use, and morphological structures at the word level. However, they encountered significantly greater difficulty with deeper grammatical structures, including article usage, verb tense, lexical appropriateness, and, especially, agreement in more complex or nested sentence constructions.

Beyond quantitative outcomes, this thesis also enabled a close examination of the datasets themselves, both in their original form and rearranged for the MultiGED shared task. In the course of aligning sentences from the development set with their source corpora, several issues were identified that may have introduced noise into both training and evaluation. Although the MultiGED format is intended to consist of isolated single sentences, many examples appear to include sequences of adjacent sentences from the original corpora, often without regard for natural sentence boundaries. Moreover, cases of duplication, inconsistent tokenization, and inaccurate sentence segmentation were observed, particularly within REALEC and MERLIN. These irregularities may originate from the nested and complex structure of the original corpora, but they also reflect inconsistencies within the shared task dataset itself.

Identifying and accounting for such issues was essential for conducting accurate error analyses and ensuring meaningful comparisons between gold annotations and model predictions. These findings highlight the need for greater transparency and preprocessing rigor in the construction of learner corpora and shared task datasets, particularly when they serve as benchmarks for multilingual grammatical error detection.

6.2 Insights from the Shared Task

The MultiGED-2023 shared task provided an essential benchmark for evaluating model performance in a multilingual GED setting. Although the XLM-RoBERTa model submitted for this thesis was a straightforward implementation, with no intention of producing a competitive system, it consistently achieved second place across all three tracks, indicating a baseline level of competitiveness relative to both monolingual and multilingual systems. Notably, it obtained an $F_{0.5}$ score of 0.771 on the Italian track and achieved a recall of 0.743 on the FCE subset for English.

It is important to stress that the model submitted in this context was a "vanilla" implementation, aimed not at optimization or performance ranking but rather at partially reproducing the experimental setup used by the top-performing team in the shared task. The primary motivation behind the submission was to establish a working baseline that could support further qualitative and error-type-oriented analyses. In this regard, the objective was not to compete with the best teams but to go beyond benchmark metrics by deepening the understanding of model behavior across error types and across languages. Despite the intentionally simple design of my "vanilla implementation", the XLM-RoBERTa models fine-tuned separately on English and Italian achieved second place in all three tracks, FCE, REALEC, and MERLIN.

The relatively modest performance observed across all submitted systems suggests that many participants may have adopted a similar exploratory approach. Given the low overall scores, even for top-ranking systems, it is plausible that the community views this shared task less as a final benchmark and more as an initial step toward more comprehensive multilingual GED research. This is particularly relevant as this is the first shared task of its kind to explicitly focus on multilingual GED in learner language.

When contextualizing the submitted system's performance relative to other participants in the MultiGED-2023 shared task, it is important to note that both top-performing teams, EliCoDe (Colla et al., 2023) and DSL-MIM-HUS (Le-Hong et al., 2023), also used XLM-RoBERTa as their base architecture. However, their fine-tuning strategies differed substantially, as mentioned in Chapter 2. EliCoDe fine-tuned the model separately for each language, whereas DSL-MIM-HUS adopted a joint multilingual fine-tuning approach. Despite their more sophisticated setups, the model submitted for this thesis, based on a straightforward fine-tuning strategy, remained competitive, typically falling between the two leading systems. This outcome reinforces the robustness of XLM-RoBERTa as a multilingual baseline and suggests that fine-tuning methodology, rather than architectural innovation, was the main driver of performance differences. Notably, DSL-MIM-HUS only outperformed the submitted system on the FCE dataset, while on both the REALEC and Italian tracks, the baseline model achieved higher scores. This suggests that even a relatively simple fine-tuning procedure can be highly effective, particularly when applied monolingually. In the case of Italian, fine-tuning on a single language appears to yield more focused learning and better generalization, likely due to reduced cross-lingual interference and the model's ability to specialize more directly on the available linguistic patterns.

6.3 Insights from the Error Analysis and Correction Generation

For English, XLM-RoBERTa exhibited a similar pattern of strengths and weaknesses, as revealed through the analysis of the FCE dataset. Subject-verb agreement errors were often misclassified when the grammatical subject was separated from the verb by intervening clauses or modifiers. Tense and aspect-related distinctions, particularly those requiring narrative cohesion or contrastive temporal framing, also proved challenging. While the model successfully identified more overt or formulaic errors, it struggled with more subtle violations involving fixed expressions, idioms, or collocational constraints.

For Italian, the error analysis revealed that the model performs well on surface-level patterns but consistently struggles with deeper morphosyntactic structures. Agreement errors,

particularly gender and number concord between determiners, nouns, and adjectives, were often missed, especially in cases involving discontinuous phrases or embedded sentence structures. Valency-related errors, such as incorrect or omitted arguments in verb constructions, also posed challenges for the model. Furthermore, article usage, which in Italian depends on complex interactions between definiteness, gender, number, and syntactic context, was frequently misclassified. Verb tense errors, which represent the error type with the lowest recall, were especially problematic. In most of these cases, the model flagged inconsistencies in the use of different tenses within sentences, such as *passato prossimo* and *imperfetto*, often reflecting real ambiguity in tense choice. However, the model also marked many verb forms as incorrect that were in fact grammatically acceptable, but which did not align with the gold annotation due to contextual expectations.

These outcomes suggest that while XLM-RoBERTa is capable of capturing many localized and clearly defined error types, it continues to show limitations when faced with structurally complex, semantically variable, or pragmatically nuanced usage patterns.

Moreover, the analysis of Qwen's generated corrections for both English and Italian revealed shared strengths and systematic weaknesses across languages and error types. The model performed reliably on surface-level errors, particularly orthographic issues, article omissions, and locally recoverable morphosyntactic violations. However, it consistently underperformed on deeper grammatical constructions that require semantic interpretation or discourse-level coherence. In English, Qwen achieved its strongest results in the False Positives and Word Order errors. By contrast, it struggled on the Replacing Verb category, failing to identify semantically inappropriate verbs and to substitute them with grammatically and contextually coherent alternatives. A similar trend was observed for Verb Tense. These patterns were echoed in the Italian results, where tense selection involving *imperfetto* and *passato prossimo*, as well as agreement in embedded structures, remained significant challenges. Taken together, these findings indicate that while Qwen can produce fluent and accurate corrections, it lacks robustness in handling more abstract or context-dependent grammatical errors.

Moving beyond the focus on the models themselves, this research has uncovered several issues inherent to both the shared task and the nature of the provided data, though not all of these constraints directly impacted the study. As previosuly mentioned in Chapter 5.2.5, one such structural constraint is the use of binary labeling in the MultiGED-2023 dataset, where each token is marked as either correct or incorrect, without any information about the type of error. While this labeling scheme was deliberately adopted by the organizers to simplify multilingual benchmarking, it inherently collapses a wide range of grammatical phenomena (e.g. missing word, unnecessary word, wrong words, etc.) into a single category. This limitation not only confuses the model, since it is unsure what to highlight, but also hinders the ability to evaluate model performance by error class or severity, and restricts interpretability when comparing linguistic patterns across languages.

However, this latter limitation, namely, the inability to evaluate performance by error type, was mitigated in the present study by returning to the original datasets, MERLIN for Italian and FCE/REALEC for English, where detailed error-type annotations are available. Although the shared task does not support this level of differentiation, integrating the original annotations made it possible to go beyond the binary labels and extract more linguistically informative insights.

Another insight regarding the shared task concerns the nature of the input and the granularity of the annotations. The models in this study were trained and evaluated at the sentence level, as per the shared task guidelines. However, closer inspection of the underlying corpora revealed that many of the gold labels were generated with access to broader discourse context, often relying on paragraph, or text-level information to determine correctness. This introduces a significant inconsistency between the input available to the model and the information used to create the ground truth. In other words, the model is asked to make predictions in isolation, while the gold labels may reflect judgments that depend on context beyond the sentence boundary. Such misalignment affects the validity of the evaluation and points to a deeper issue: either the data should consist of full texts or paragraphs, with context-aware labeling and model input, or it should be limited to self-contained sentences with labels that are interpretable without

external context. The current setup, which extracts random sentences from larger texts while retaining context-dependent labels, risks penalizing models for failing to infer information they were never given.

Finally, issues were also identified regarding the availability and reliability of multiple reference corrections. The MERLIN dataset for Italian is frequently cited as offering two levels of correction: a minimal grammatical revision (TH1) and a more comprehensive, stylistic or sociolinguistic alternative (TH2). However, in practice, this distinction is not consistently evident. In the subset of sentences analyzed during both the error analysis and Qwen generation phases, the TH2 corrections rarely diverged meaningfully from TH1. In many cases, they were identical or only minimally modified. As a result, TH2 offered limited added value for evaluating model sensitivity to correction diversity or nuance in language use, exposing a gap between the theoretical richness of the dataset and its practical utility for multi-reference evaluation.

Taken together, these observations point to a broader conclusion: the evaluation of grammatical error detection models cannot be reduced to numerical scores. In many instances, XLM-RoBERTa correctly identified issues that were not captured in the gold annotations, highlighting inconsistencies in both the annotation schemes and the evaluation protocols of the shared task. This shows that grammatical error detection is not merely a technical problem, but a linguistically layered one which cannot be fully addressed through overly simplistic solutions. For instance, frameworks based on a single correction, or overly rigid label sets, tend to obscure the complexity of learner language and the range of possible valid revisions. Even when multiple corrections are available, as in the case of MERLIN, they are often ignored or indistinct in practice. Furthermore, some model outputs, though not aligned with the reference, offered valid alternative corrections, underscoring the need for more flexible, context-sensitive, and linguistically informed approaches to both annotation and evaluation. Ultimately, this study reinforces the importance of critically interrogating both model behavior and the assumptions built into the data and metrics used to assess it.

6.4 Limitations of the Study

Several methodological limitations of this study must be acknowledged.

First, as mentioned above, the selected monolingual models represent both small and large-scale architectures. Unsurprisingly, the larger models achieved substantially stronger results. However, all multilingual models were evaluated only in their base configurations. This raises the possibility that scaling up multilingual architectures could help narrow the performance gap between monolingual and multilingual systems.

A further limitation concerns the experimental design: evaluations were not consistently conducted across multiple random seeds, which is important because results can vary depending on the random initialization of parameters. Similarly, the number of epochs was fixed at three, which may not have allowed all models to reach optimal convergence. In addition, memory constraints in the CPU-based setup led to training failures for some multilingual and large monolingual models, requiring a reduction of the batch size from 16 to 8, which may have affected training dynamics and final model performance. These factors may affect the robustness and reproducibility of the reported findings.

Another limitation lies in the use of the Qwen model to generate corrections. Since the main goal of this thesis was not to optimize correction quality, but rather to explore the diversity of possible corrections beyond those included in the original datasets, prompting strategies were not extensively tuned.

Furthermore, while this study focused on English and Italian, two well-resourced languages, similar pipelines may face greater challenges in lower-resource settings, where annotated data and model support are limited.

Finally, the last limitation lies in the scope of the error types selected for the qualitative analysis. The categories examined in this study were chosen based on their frequency, reflected in recall scores, and their relevance to grammatical phenomena that often allow for multiple plausible interpretations or corrections. This targeted selection enabled meaningful

cross-linguistic comparisons between English and Italian and allowed to analyze model behavior. However, the selection does not represent the full range of error categories present in the data, and therefore does not offer a comprehensive picture of the model's performance across the entire error taxonomy. Performing an exhaustive error-type analysis would have been prohibitively time-consuming, particularly given the structural and typological differences between English and Italian and the differing granularity of their annotation schemes. Nevertheless, the fact that the taxonomies used across MERLIN, FCE, and REALEC are broadly comparable represents a solid foundation for future multilingual evaluations that aim to be more exhaustive in scope.

It must also be said that the availability and distribution of available data for error types limits my ability to fully generalize my findings.

These limitations underscore the importance of future work that not only expands the scope of error categories and correction references, but also systematically examines prompting strategies, improves reproducibility, and addresses cross-linguistic generalizability.

6.5 Future Work

The results and analyses presented in this thesis open several promising directions for future research in multilingual grammatical error detection. One immediate extension concerns the scaling of multilingual models. The current study relied on base versions of XLM-RoBERTa and mBERT, while monolingual models were explored at both base and large scales. As observed, larger monolingual models performed notably better, particularly for Italian. Future experiments should investigate whether similarly scaling multilingual models, such as XLM-RoBERTa-large, could close the performance gap or even outperform monolingual systems when trained on sufficiently rich learner data.

Another avenue involves enriching the model input with broader contextual information. A recurring limitation in both detection and correction was the mismatch between sentence-level inputs and context-dependent gold labels. Integrating paragraph- or document-level context into the model architecture or in the data structure, may allow models to reason over tense continuity, discourse coherence, and pragmatic appropriateness more effectively.

Improvements to dataset quality and preprocessing also remain crucial. As highlighted in this study, both the shared task datasets and the original corpora contain structural inconsistencies, such as incorrect sentence segmentation, duplicated examples, and ambiguous boundary definitions. Future shared tasks would benefit from stricter dataset validation procedures, clearer sentence delimitation, and metadata that explicitly indicates the scope of each annotation. Moreover, increasing transparency in preprocessing pipelines could help standardize multilingual evaluation settings and facilitate reproducibility across studies.

The evaluation framework itself could also be expanded. While this thesis partially addressed the limitations of binary token-level labeling by restoring access to original error types, future work could pursue a more fine-grained, multi-label classification setup from the outset. This would allow models not only to detect the presence of an error, but also to specify its type improving interpretability and pedagogical usefulness. Importantly, future evaluation schemes could be designed to accommodate two tiers of analysis: a simplified version based on binary correctness for low-resource or baseline systems, and a more sophisticated error-type-aware version for systems capable of deeper linguistic reasoning. Such a dual framework would support broader participation in shared tasks while also encouraging more nuanced and informative evaluations in advanced settings.

An especially promising direction lies in systematically linking discriminative and generative model outputs. The question of whether a token-level prediction that deviates from the gold label aligns with an alternative, linguistically valid correction is central to both robustness and evaluation. Future research could investigate methods for automatically mapping token-level predictions from discriminative models (e.g., XLM-RoBERTa) to the corrections generated by large language models (e.g., Qwen or ChatGPT). Techniques such as alignment via Levenshtein edit distance, span-based error extraction, or embedding similarity could be explored to

associate classification decisions with specific correction hypotheses.

Additionally, the role of generative models in GED and GEC remains underexplored. While the present study used Qwen to generate candidate corrections and evaluated them qualitatively, future work could leverage generation more systematically. It would be crucial to have access to datasets that include multiple gold references per sentence, capturing a range of plausible corrections rather than a single canonical answer. This remains a significant limitation even in corpora such as MERLIN, where alternative correction layers are provided in theory but rarely diverge in practice. Moreover, generation could also be used to create synthetic data to augment error types that are poorly represented in existing corpora.

Generative approaches could also benefit from the incorporation of memory mechanisms or broader discourse modeling to capture context dependencies that influence correction choices. This is particularly relevant for phenomena such as tense and aspect, where the appropriateness of a correction often depends on events or cues outside of the sentence itself. Future work could test whether models equipped with memory modules are better able to produce contextually coherent corrections, especially in narrative texts or learner compositions involving temporal progression.

In sum, future work should aim to integrate larger models, richer context, cleaner data, better alignment between detection and correction, and broader coverage of error types. Together, these improvements can move the field toward more robust, interpretable, and pedagogically meaningful multilingual GED systems.

Chapter 7

Conclusion

This thesis set out to investigate how multilingual and monolingual transformer-based models handle grammatical error detection in texts produced by learners of English and Italian as second languages, particularly in the context of language exams. The focus was on cases that could present ambiguity due to multiple plausible correction options, and, emerging during the course of the analysis, on instances marked by annotation inconsistencies. Through a combination of quantitative evaluation and qualitative error analysis, the study examined model behavior across a wide range of error types and linguistic contexts, revealing both the capabilities and limitations of current systems.

To address research question 1, four monolingual models (DistilBERT and BERT-large-cased for English, and bert-base-italian-uncased and bert-base-italian-xxl-cased for Italian) and three multilingual models (bert-base-multilingual-uncased, bert-base-multilingual-cased, and XLM-RoBERTa-base) were fine-tuned for the grammatical error detection task defined for the MultiGED 2023 shared task. The multilingual models, although capable of joint training across languages, were deliberately fine-tuned separately for English and Italian. The experimental results demonstrated that multilingual models such as XLM-RoBERTa offer robust cross-lingual performance, but do not consistently outperform large monolingual baselines when fine-tuned separately on a single language. The submitted XLM-RoBERTa model, despite being a baseline implementation, performed competitively across all tracks in the MultiGED-2023 shared task.

Extracting the original error type labels from the FCE, REALEC and MERLIN datasets was crucial to answer research question 3. It can be said, in fact, that a core finding of this work is that model performance is highly dependent on the nature of the error. While surface-level patterns, such as orthographic issues and common morphological errors, are handled effectively, deeper grammatical phenomena like article usage, tense, valency, and agreement remain challenging, especially when embedded in complex or ambiguous constructions. These difficulties were evident in both the token-level classification results and the generative corrections, suggesting that further advances in modeling linguistic structure and discourse context are needed.

The error analysis, both on XLM-RoBERTa's predictions and Qwen's generated corrections, was essential to answering research question 2 and 4. The qualitative analysis, which included targeted generation and detailed comparison with gold annotations, provided new insights into annotation inconsistencies, the limitations of binary evaluation, and the potential of generative models like Qwen to offer linguistically valid corrections even when diverging from the gold standard. In many cases, both the discriminative outputs and generative outputs highlighted alternative plausible corrections, challenging the assumption that gold annotations always capture the full range of acceptable revisions. This suggests that generation can complement classification, especially when evaluating ambiguous inputs. It also supports the case for expanding the evaluation framework to account for multiple valid outputs. This underscores the need for more flexible and nuanced evaluation frameworks that go beyond binary labels and adapt to linguistic variability.

Finally, this study identified several practical limitations in existing datasets, ranging from tokenization and segmentation issues to duplicated examples, and emphasized the importance

of rigorous preprocessing and documentation in multilingual learner corpora. It also outlined directions for future research, including better alignment between discriminative and generative outputs, multi-label classification setups, and the integration of broader context and memory mechanisms.

In sum, this thesis contributes to a more linguistically grounded understanding of how transformer-based models handle ambiguity and variation in GED, particularly in low-resource and multilingual contexts. By bridging classification and generation, and by introducing interpretability together with performance, it can be a starting point for more robust and pedagogically centered grammatical error detection systems.

Appendix A

Prompting strategy for QWEN

This appendix documents the prompts used to generate sentence-level grammatical corrections using QWEN. The objective was to obtain plausible, tokenized corrections that reflect minimal grammatical adjustments while preserving sentence structure as much as possible. Several variants were tested, reflecting different levels of constraint on token length, edit minimality, and output formatting.

Prompt 1: Minimal edits, fixed structure

```
Here is a tokenized sentence: {sentence_str}
Provide 3 corrected versions of this sentence, tokenized as well, try to keep
    ↪ the changes as little as possible.
Separate the corrections by new lines.
Each correction should be a sequence of tokens separated by spaces.
Example output:
Il gatto sta dormendo .
Un gatto sta dormendo .
I gatti stanno dormendo .
```

This initial prompt aims to constrain the model to make minimal changes, with an explicit example encouraging grammatical variety. However, it does not enforce token length preservation, occasionally leading to structural divergence from the input.

Prompt 2: Enforced token count

```
Here is a tokenized sentence: {sentence_str}
Provide 3 corrected versions of this sentence, tokenized as well,
each with the same number of tokens as the original.
Separate the corrections by new lines.
Each correction should be a sequence of tokens separated by spaces.
Example output:
The cat is sleeping .
A cat is sleeping .
The cats are sleeping .
```

This version introduces a stricter constraint by requiring each corrected sentence to contain the same number of tokens as the original. This was especially useful when alignment or direct comparison with gold annotations was necessary, although it slightly limited grammatical flexibility.

Prompt 3: Variable number of outputs, fallback

```
Here is a tokenized sentence: {sentence_str}
Provide up to 5 corrected versions of this sentence, tokenized as well, try to
    ↪ keep the changes as little as possible.
Separate the corrections by new lines.
Each correction should be a sequence of tokens separated by spaces.
Example input: gatto sta dormendo .
Example output:
Il gatto sta dormendo .
Un gatto sta dormendo .
If you have no corrections give me an empty list.
```

This prompt increased the maximum number of outputs while introducing a fallback condition for well-formed input. The option to return an empty list was meant to reduce unnecessary rephrasings when the input was already grammatical, but in practice, models often ignored this instruction.

Prompt 4: Role-based instruction

```
You are a careful Italian language editor.

Given a tokenized sentence, return up to three corrected versions. Each version
    ↪ should:
- Be minimally corrected (grammar/lexicon only)
- Be tokenized (space-separated words)
- Be written on a separate line
- No explanations

Input: {sentence_str}
Output:
```

This second to last version adopts a role-based instruction style, which often improves output consistency with instruction-following models. It also removes redundancy by dropping examples, focusing instead on declarative constraints. This version generally produced the most linguistically plausible and appropriately scoped corrections.

Overall, prompt design played a critical role in shaping the quality and consistency of generated corrections. Variations in phrasing, constraints, and examples significantly affected model behavior, and careful tuning was necessary to ensure outputs aligned with the goals of this thesis.

Appendix B

Results: Aligned Datasets

B.1 English (FCE)

Table B.1 presents the results on the preprocessed version of the FCE dataset, which, as mentioned above, slightly differs from the original one (Table 4.2). Overall, performance on the majority label *c* remains stable across both versions, with $F_{0.5}$ scores consistently ranging between 0.94 and 0.95 across all models. This suggests that the filtering process did not significantly affect the models' ability to identify correctly used tokens.

Even for the minority class *i*, no substantial changes are observed. All models show no change in precision. Recall, on the other hand, decreases marginally for bert-base-multilingual-uncased which goes from 0.42 to 0.41.

Model	**Label c**				**Label i**			
	P	R	$F_{0.5}$	supp.	P	R	$F_{0.5}$	supp.
distilbert-base-uncased	**0.93**	**0.98**	**0.94**	**31254**	**0.71**	**0.40**	**0.61**	**3449**
bert-large-cased	0.94	0.97	0.95	31254	0.71	0.49	0.65	3449
bert-base-multilingual-cased	0.93	0.98	0.94	31254	0.71	0.40	0.61	3449
bert-base-multilingual-uncased	0.93	0.98	0.94	31254	0.72	0.41	0.63	3449
xlm-roberta-base	**0.94**	**0.98**	**0.94**	**31254**	**0.74**	**0.43**	**0.65**	**3449**

Table B.1: Processed-FCE: Precision, Recall, and $F_{0.5}$ scores for label *c* and label *i*.

As visible from Figure B.1, the number of true positives for both labels decreases slightly in the pre-processed version due to the reduced number of sentences. However, the class balance of predictions remains consistent across model. For instance, in the original dataset, BERT-large-cased correctly identified 1,703 tokens with label *i* and misclassified 1,757, resulting in a near 1:1 ratio. In the pre-processed version, it predicts 1694 true positives and 1,755 false negatives, indicating only a slight shift in balance which can suggest that the removed examples did not introduce major distortions.

XLM-RoBERTa continues to perform best in identifying incorrect tokens. Its true positives for *i* decrease marginally from 1,497 to 1489, and false negatives from 1,963 to 1960, suggesting stable performance despite the dataset reduction. Similarly, DistilBERT-base-uncased exhibits a small drop in true positives, from 1,388 to 1,382, with false negatives decreasing slightly from 2,072 to 2,067. These trends confirm that while absolute numbers are slightly lower, which of course is understandable, the overall class distribution and relative performance patterns are preserved, validating the comparability of results between the original and filtered datasets.

	Pred c	Pred i
True c	30691	563
True i	2067	1382

(a) DistilBERT-base-uncased

	Pred c	Pred i
True c	30579	675
True i	1755	1694

(b) BERT-large-cased

	Pred c	Pred i
True c	30694	560
True i	2053	1396

(c) m-BERT-cased

	Pred c	Pred i
True c	30709	545
True i	2002	1447

(d) m-BERT-uncased

	Pred c	Pred i
True c	30755	499
True i	1960	1489

(e) XLM-RoBERTa

Figure B.1: Confusion matrices for five models on the Pre-Processed FCE.

B.2 English (REALEC)

As visibile, in Table B.2, also for the preprocessed REALEC, as expected, the removal of certain samples in the preprocessed set did not significantly impact the models' ability to classify the majority label *c*, with $F_{0.5}$ scores remaining stable at 0.93 or 0.94 across all models. Even for the minority class *i*, the results are remarkably consistent with the original dataset, with changes in $F_{0.5}$ rarely exceeding 0.01. The best-performing model on class *i* remains xlm-roberta-base, maintaining an $F_{0.5}$ of 0.45, which is identical to its original result.

Model	**Label c**				**Label i**			
	P	R	$F_{0.5}$	supp.	P	R	$F_{0.5}$	supp.
bert-large-cased	**0.93**	**0.95**	**0.94**	**78275**	**0.45**	**0.40**	**0.44**	**8087**
distilbert-base-uncased	0.93	0.96	0.93	78275	0.47	0.34	0.44	8087
bert-base-multilingual-cased	0.93	0.96	0.93	78275	0.47	0.34	0.44	8087
bert-base-multilingual-uncased	0.93	0.96	0.93	78275	0.48	0.34	0.44	8087
xlm-roberta-base	**0.93**	**0.96**	**0.94**	**78275**	**0.49**	**0.35**	**0.45**	**8087**

Table B.2: Processed-REALEC: Precision, Recall, and $F_{0.5}$ scores for label *c* and label *i*.

Even by taking into consideration the confusion matrices for the processed REALEC dataset, as in Figure B.2, in comparison with the original REALEC dataset for the MultiGED-2023 shared task in Figure 4.2, it is noticeable that across both versions, model behavior remains largely consistent, with all systems correctly classifying the vast majority of tokens labeled as *c*. However, some variations can be observed, particularly in the number of correctly predicted *i* tokens. For instance, BERT-large-cased sees a slight drop in true positives for class *i* (from 3,265 to 3,257) and a corresponding reduction in false negatives (from 4,838 to 4,830), indicating nearly identical performance despite a smaller test set. Similarly, XLM-RoBERTa improves its true positives slightly from 2,919 to 2,909, while also decreasing false negatives from 5,184 to 5,178.

For other models like DistilBERT and the multilingual BERT variants, the differences are minimal and largely proportional to the reduced number of total examples in the pre-processed

dataset. For instance, m-BERT-cased decreases from 2,802 to 2,795 true positives on class *i*, while false negatives remain around 5,300 in both cases. Interestingly, despite a reduction of 52 instances in the pre-processed REALEC dataset, the number of true positives for label *i* remains almost unchanged across all models. This stability suggests that the excluded examples were potentially introducing noise rather than useful information. As such, the removal of these cases may have led to a cleaner evaluation set, resulting in more reliable estimates of model performance on well-formed examples.

	Pred c	**Pred i**
True c	75218	3057
True i	5282	2805

(a) DistilBERT-base-uncased

	Pred c	**Pred i**
True c	74423	3852
True i	4830	3257

(b) BERT-large-cased

	Pred c	**Pred i**
True c	75229	3046
True i	5292	2795

(c) m-BERT-cased

	Pred c	**Pred i**
True c	75314	2961
True i	5332	2755

(d) m-BERT-uncased

	Pred c	**Pred i**
True c	75273	3002
True i	5178	2909

(e) XLM-RoBERTa

Figure B.2: Confusion matrices for five models on the Pre-Processed REALEC.

B.3 Italian (MERLIN)

Table B.3 shows the performance of all models on the preprocessed version of the MERLIN dataset. Consistent with the original results in Table 4.6, is also the performance of the pre-processed MERLIN, as noticeable in Table B.3, indicating that the preprocessing step did not have a significant impact on performance for this class.

For the minority label *i*, the results are stable. $F_{0.5}$ scores are almost unchanged. Notably, multilingual and monolingual models maintain their relative rankings, with bert-base-italian-xxl-cased achieving the highest score (0.80), and bert-base-multilingual-uncased remaining the lowest (0.60 in the preprocessed version, previously 0.59).

Model	**Label c**				**Label i**			
	P	R	$F_{0.5}$	supp.	P	R	$F_{0.5}$	supp.
bert-base-italian-uncased	0.92	0.98	0.93	7678	0.77	0.44	0.67	1172
bert-base-italian-xxl-cased	**0.94**	**0.98**	**0.95**	**7678**	**0.86**	**0.64**	**0.80**	**1172**
bert-base-multilingual-cased	0.92	0.97	0.93	7678	0.78	0.48	0.69	1172
bert-base-multilingual-uncased	**0.90**	**0.97**	**0.92**	**7678**	**0.72**	**0.35**	**0.60**	**1172**
xlm-roberta-base	0.93	0.98	0.94	7678	0.83	0.52	0.74	1172

Table B.3: Processed-MERLIN: Precision, Recall, and $F_{0.5}$ scores for label *c* and label *i*.

In Figure B.3, one can see that for all models, the number of correctly predicted majority-class tokens (label *c*) remains very high, with true positives always above 7,500.

For the minority class *i*, performance is slightly reduced in absolute terms due to fewer examples, but the ratio of true positives to false negatives remains stable. For example, XLM-RoBERTa correctly identifies 637 tokens in the original version and 616 in the pre-processed one, while false negatives drop from 574 to 556. Similarly, bert-base-italian-xxl-cased shows almost identical results: 768 true positives in the original and 754 in the filtered version.

	Pred c	**Pred i**
True c	7527	151
True i	645	527

(a) BERT-base-Italian-uncased

	Pred c	**Pred i**
True c	7560	118
True i	418	754

(b) BERT-XXL-Italian-cased

	Pred c	**Pred i**
True c	7518	160
True i	603	569

(c) m-BERT-cased

	Pred c	**Pred i**
True c	7518	160
True i	752	420

(d) m-BERT-uncased

	Pred c	**Pred i**
True c	7559	119
True i	556	616

(e) XLM-RoBERTa

Figure B.3: Confusion matrices for five models on the Pre-Processed MERLIN.

Appendix C

Error Type Performance Comparison

C.1 Results (FCE): Error Type Performance comparison between BERT-large-cased and XLM-RoBERTa

Error Type	**Gold**	**TP (B)**	**TP (R)**	**FN (B)**	**FN (R)**	**Recall (B)**	**Recall (R)**
Incorrect Form Noun Plur.	11	11	10	0	1	1.00	0.91
Derived Determiner	4	4	3	0	1	1.00	0.75
Incorrect Quantifier	2	2	2	0	0	1.00	1.00
Derived Verb	7	7	6	0	1	1.00	0.86
Form Adverb	4	4	3	0	1	1.00	0.75
Quantifier Agreement	1	1	0	0	1	1.00	0.00
Incorrect Anaphoric	2	2	2	0	0	1.00	1.00
Missing Verb	1	1	1	0	0	1.00	1.00
Incorrect Verb Inflection	30	29	30	1	0	0.97	1.00
Derived Pronoun	15	14	14	1	1	0.93	0.93
Spelling	331	304	304	27	27	0.92	0.92
Noun Countability	9	8	9	1	0	0.89	1.00
Determiner Agreement	7	6	6	1	1	0.86	0.86
Derived Noun	34	26	23	8	11	0.76	0.68
Confusion Spelling	50	38	31	12	19	0.76	0.62
Missing Determiner	8	6	6	2	2	0.75	0.75
Verb Agreement	55	40	41	15	14	0.73	0.75
Derived Adjective	46	33	30	13	16	0.72	0.65
Derived Adverb	23	16	15	7	8	0.70	0.65
American Spelling	21	14	11	7	10	0.67	0.52
Adjective Form	3	2	2	1	1	0.67	0.67
Unnecessary Determiner	8	5	5	3	3	0.62	0.62
Replacing Punctuation	241	145	141	96	100	0.60	0.59
Pronoun Agreement	5	3	1	2	4	0.60	0.20
Verb Form	106	62	54	44	52	0.58	0.51
Noun Form	57	33	28	24	29	0.58	0.49
Replacing Preposition	205	115	106	90	99	0.56	0.52
Incorrect Adjective	9	5	3	4	6	0.56	0.33
Noun Agreement	69	37	33	32	36	0.54	0.48
UNKNOWN label	559	260	214	299	345	0.47	0.38

Continued on next page

Error Type	Gold	TP (B)	TP (R)	FN (B)	FN (R)	Recall (B)	Recall (R)
Unnecessary Punctuation	24	10	6	14	18	0.42	0.25
Missing Pronoun	5	2	3	3	2	0.40	0.60
Replacing Adjective	48	19	12	29	36	0.40	0.25
Idiom	53	19	15	34	38	0.36	0.28
Verb Tense	235	84	73	151	162	0.36	0.31
Word Order	327	107	99	220	228	0.33	0.30
Replacing Noun	101	32	24	69	77	0.32	0.24
Replacing Verb	205	64	46	141	159	0.31	0.22
Replacing Quantifier	13	4	2	9	11	0.31	0.15
Incorrect Negation	23	7	5	16	18	0.30	0.22
Replacing Determiner	41	12	5	29	36	0.29	0.12
Replacing Pronoun	55	16	7	39	48	0.29	0.13
Replacing Conjunction	19	5	0	14	19	0.26	0.00
Replacing	150	39	25	111	125	0.26	0.17
Missing	4	1	2	3	2	0.25	0.50
Inappropriate Register	8	2	3	6	5	0.25	0.38
Unnecessary Verb	4	1	1	3	3	0.25	0.25
Argument Structure	91	19	9	72	82	0.21	0.10
Unnecessary/Redundant	5	1	0	4	5	0.20	0.00
Replacing Adverb	59	11	10	48	49	0.19	0.17
Missing Punctuation	28	5	7	23	21	0.18	0.25
Wrong Quantifier	6	1	1	5	5	0.17	0.17
Derived Preposition	1	0	0	1	1	0.00	0.00
Unnecessary Pronoun	6	0	0	6	6	0.00	0.00
Unnecessary Conjunction	3	0	0	3	3	0.00	0.00
Missing Preposition	1	0	0	1	1	0.00	0.00
Determiner Form	6	0	0	6	6	0.00	0.00
Unnecessary Adverb	3	0	0	3	3	0.00	0.00
Unnecessary Preposition	2	0	0	2	2	0.00	0.00
TOTAL	**3449**	**1694**	**1489**	**1755**	**1960**	**0.49**	**0.43**

Table C.1: Recall per error type for BERT-cased and XLM-RoBERTa models on the FCE dataset

C.2 Results (REALEC): Error Type Performance comparison between BERT-large-cased and XLM-RoBERTa

Error Code	Gold	TP (B)	TP (R)	FN (B)	FN (R)	Recall (B)	Recall (R)
Adj_as_collective	1	1	1	0	0	1.0	1.0
Adverbs	1	1	1	0	0	1.0	1.0
Vocabulary	1	1	1	0	0	1.0	1.0
Countable_uncountable	9	8	7	1	2	0.89	0.78
Spelling	1503	1327	1268	176	235	0.88	0.84
Agreement_errors	184	128	124	56	60	0.7	0.67
Tense_form	74	48	36	26	38	0.65	0.49
Derivation	8	5	3	3	5	0.62	0.38
Capitalisation	77	48	41	29	36	0.62	0.53
Category_confusion	122	76	59	46	63	0.62	0.48

Continued on next page

Error Code	**Gold**	**TP (B)**	**TP (R)**	**FN (B)**	**FN (R)**	**Recall (B)**	**Recall (R)**
Numerals	52	28	29	24	23	0.54	0.56
Formational_affixes	32	17	14	15	18	0.53	0.44
Noun_number	206	103	101	103	105	0.5	0.49
Lack_par_constr	6	3	3	3	3	0.5	0.5
Adjectives	6	3	3	3	3	0.5	0.5
Possessive	35	17	14	18	21	0.49	0.4
Articles	851	411	381	440	470	0.48	0.45
Voice	56	26	18	30	38	0.46	0.32
Noun_inf	5	2	1	3	4	0.4	0.2
note	5	2	2	3	3	0.4	0.4
Nouns	5	2	2	3	3	0.4	0.4
Pronouns	23	9	9	14	14	0.39	0.39
suggestion	18	7	9	11	9	0.39	0.5
Confusion_of_structures	38	14	12	24	26	0.37	0.32
Prepositions	248	89	80	159	168	0.36	0.32
Determiners	48	17	16	31	32	0.35	0.33
Ref_device	114	36	21	78	93	0.32	0.18
lex_item_choice	782	244	187	538	595	0.31	0.24
Prepositional_adjective	13	4	2	9	11	0.31	0.15
Absence_comp_sent	215	66	49	149	166	0.31	0.23
Quantifiers	7	2	1	5	6	0.29	0.14
lex_part_choice	159	45	36	114	123	0.28	0.23
Prepositional_noun	32	9	6	23	26	0.28	0.19
Verb_pattern	97	27	19	70	78	0.28	0.2
Infinitive_constr	8	2	4	6	4	0.25	0.5
Redundant_comp	117	28	18	89	99	0.24	0.15
Tense_choice	368	80	66	288	302	0.22	0.18
Comparative_constr	19	4	3	15	16	0.21	0.16
Word_choice	227	46	34	181	193	0.2	0.15
Conjunctions	32	6	4	26	28	0.19	0.12
Absence_explanation	177	33	26	144	151	0.19	0.15
Participial_constr	40	7	9	33	31	0.17	0.23
Compound_word	23	4	4	19	19	0.17	0.17
Linking_device	74	12	14	62	60	0.16	0.19
Modals	26	4	5	22	21	0.15	0.19
Word_order	507	77	56	430	451	0.15	0.11
	455	62	58	393	397	0.14	0.13
Inappropriate_register	105	13	10	92	95	0.12	0.1
Negation	22	2	3	20	19	0.09	0.14
Coherence	52	4	3	48	49	0.08	0.06
Relative_clause	144	9	8	135	136	0.06	0.06
Punctuation	644	38	28	606	616	0.06	0.04
Verbs	1	0	0	1	1	0.0	0.0
Discourse	7	0	0	7	7	0.0	0.0
Comparison_degree	5	0	0	5	5	0.0	0.0
Prepositional_adv	1	0	0	1	1	0.0	0.0
TOTAL	**8087**	**3257**	**2909**	**4830**	**5178**	**0.40**	**0.36**

Table C.2: Recall per error type for BERT-cased and XLM-RoBERTa models on the REALEC dataset

C.3 Results (MERLIN): Error Type Performance comparison between BERT-base-italian-XXL-cased and XLM-RoBERTa

Error Type	Gold	TP (B)	TP (R)	FN (B)	FN (R)	Recall (B)	Recall (R)
Vocabulary_Word-FS-Denotation	1	1	0	0	1	1.00	0.00
Coherence_Content-Jump	1	1	1	0	0	1.00	1.00
Grammar_Verb-Formation	17	15	11	2	6	0.88	0.65
Orthography_Capitalization	43	37	36	6	7	0.86	0.84
Orthography_Grapheme	154	132	108	22	46	0.86	0.70
Grammar_Inexistent-Inflection	26	22	14	4	12	0.85	0.54
Orthography_Word-Boundary	10	8	7	2	3	0.80	0.70
Grammar_Wrong-Inflection	52	40	29	12	23	0.77	0.56
Grammar_Main-Verb	28	21	16	7	12	0.75	0.57
Orthography_Apostrophe	4	3	2	1	2	0.75	0.50
Grammar_Reflexive-Pronoun	4	3	3	1	1	0.75	0.75
Pragmatics_Request	4	3	2	1	2	0.75	0.50
Grammar_Preposition	82	58	49	24	33	0.71	0.60
Orthography_Punctuation	62	42	38	20	24	0.68	0.61
V_Formulaic-Sequence-Form	3	2	1	1	2	0.67	0.33
Sociolin_Variation	3	2	2	1	1	0.67	0.67
UNK	295	193	162	102	133	0.65	0.55
Grammar_Part-Of-Speech	22	13	10	9	12	0.59	0.45
Grammar_Article	72	42	30	30	42	0.58	0.42
Grammar_Clitic	14	7	7	7	7	0.50	0.50
Grammar_Conjunction	28	14	11	14	17	0.50	0.39
Intelligibility_Sentence	2	1	1	1	1	0.50	0.50
Grammar_Negation	4	2	1	2	3	0.50	0.25
Intelligibility_Text	79	38	31	41	48	0.48	0.39
Grammar_Agreement	15	7	5	8	10	0.47	0.33
Vocabulary_Formulaic-Sequence	13	6	4	7	9	0.46	0.31
Sociolin_Text-Type-Specificity	17	7	9	10	8	0.41	0.53
Grammar_Verb-Valency	28	11	10	17	18	0.39	0.36
Grammar_Verb	37	10	5	27	32	0.27	0.14
Grammar_Word-Order	51	13	11	38	40	0.25	0.22
Coherence_Connector-Accuracy	1	0	0	1	1	0.00	0.00
TOTAL	**1172**	**754**	**616**	**418**	**556**	**0.64**	**0.53**

Table C.3: Recall per error type for bert-large-italian-cased and XLM-RoBERTa models on the FCE dataset

References

A. Akbik, D. Blythe, and R. Vollgraf. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th international conference on computational linguistics*, pages 1638–1649, 2018.

S. Bell, H. Yannakoudakis, and M. Rei. Context is key: Grammatical error detection with contextual word representations. In H. Yannakoudakis, E. Kochmar, C. Leacock, N. Madnani, I. Pilán, and T. Zesch, editors, *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 103–115, Florence, Italy, Aug. 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4410. URL `https://aclanthology.org/W19-4410/`.

P. Bhaskar, A. Ghosh, S. Pal, and S. Bandyopadhyay. Detection and correction of preposition and determiner errors in English: HOO 2012. In J. Tetreault, J. Burstein, and C. Leacock, editors, *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 201–207, Montréal, Canada, June 2012. Association for Computational Linguistics. URL `https://aclanthology.org/W12-2023/`.

A. Boyd, J. Hana, and A. Rosen. The merlin corpus: Learner language and the cefr. In *Proceedings of LREC*, pages 1281–1288, 2014.

C. Bryant, M. Felice, O. Andersen, and T. Briscoe. The bea-2019 shared task on grammatical error correction. In *Proceedings of the 14th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–75, 2019.

J. Burstein, M. Chodorow, and C. Leacock. Automated essay evaluation: The criterion online writing service. *Ai magazine*, 25(3):27–27, 2004.

Cambridge University Press and University of Cambridge. Write & Improve. `https://writeandimprove.com/`, n.d. Accessed: 2025-06-02.

K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*, 2020.

D. Colla, M. Delsanto, and E. Di Nuovo. EliCoDe at MultiGED2023: fine-tuning XLM-RoBERTa for multilingual grammatical error detection. In D. Alfter, E. Volodina, T. François, A. Jönsson, and E. Rennes, editors, *Proceedings of the 12th Workshop on NLP for Computer Assisted Language Learning*, pages 24–34, Tórshavn, Faroe Islands, May 2023. LiU Electronic Press. URL `https://aclanthology.org/2023.nlp4call-1.3/`.

A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov. Unsupervised cross-lingual representation learning at scale. In D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.747. URL `https://aclanthology.org/2020.acl-main.747/`.

D. Dahlmeier and H. T. Ng. Building a large annotated corpus of learner english: The nucle corpus of learner english. In *Proceedings of the eighth workshop on Innovative use of NLP for building educational applications*, pages 22–31, 2013.

D. Dahlmeier, H. T. Ng, and E. J. F. Ng. NUS at the HOO 2012 shared task. In J. Tetreault, J. Burstein, and C. Leacock, editors, *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 216–224, Montréal, Canada, June 2012. Association for Computational Linguistics. URL https://aclanthology.org/W12-2025/.

R. Dale and A. Kilgarriff. Helping our own: The HOO 2011 pilot shared task. In C. Gardent and K. Striegnitz, editors, *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 242–249, Nancy, France, Sept. 2011. Association for Computational Linguistics. URL https://aclanthology.org/W11-2838/.

DBMDZ. dbmdz/berts: Pretrained bert models for german and italian. https://github.com/dbmdz/berts, 2020. Accessed: 2025-06-20.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, and T. Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423/.

B. Garofolin, M. Miozzo, et al. Analisi dell'errore nell'acquisizione dell'italiano in un contesto ls e in contesti l2. *EDUCAZIONE LINGUISTICA LANGUAGE EDUCATION*, 5(3):433–452, 2016.

P. Gaurav. Mastering classification metrics: A beginner's guide [part 2: F1, f0.5, and f2 scores], Mar. 2023. URL https://medium.com/@prateekgaurav/mastering-classification-metrics-a-beginners-guide-part-2-f1-f0-5-and-f2-scores-154d5c729085 Accessed: 2025-06-02.

A. R. Golding. A bayesian hybrid method for context-sensitive spelling correction. *arXiv preprint cmp-lg/9606001*, 1996.

S. Granger. The computer learner corpus: a versatile new source of data for sla research. In *Learner English on computer*, pages 3–18. Routledge, 2014.

M. Kaneko and M. Komachi. Multi-head multi-layer attention to deep language representations for grammatical error detection, 2019. URL https://arxiv.org/abs/1904.07334.

S. Kasewa, P. Stenetorp, and S. Riedel. Wronging a right: Generating better errors to improve grammatical error detection. In E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4977–4983, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1541. URL https://aclanthology.org/D18-1541/.

E. Kuzmenko and A. Kutuzov. Russian error-annotated learner english corpus: a tool for computer-assisted language learning. In *Proceedings of the third workshop on NLP for computer-assisted language learning*, pages 87–97, 2014.

P. Le-Hong, T. Q. Ngo, and T. M. H. Nguyen. Two neural models for multilingual grammatical error detection. In D. Alfter, E. Volodina, T. François, A. Jönsson, and E. Rennes, editors, *Proceedings of the 12th Workshop on NLP for Computer Assisted Language Learning*, pages 40–44, Tórshavn, Faroe Islands, May 2023. LiU Electronic Press. URL https://aclanthology.org/2023.nlp4call-1.5/.

C. Leacock, M. Chodorow, M. Gamon, and J. Tetreault. *Automated grammatical error detection for language learners*. Morgan & Claypool Publishers, 2014.

S. Lee and K. J. Lee. A comparison of grammatical error detection techniques for an automated english scoring system. *Journal of Advanced Marine Engineering and Technology (JAMET)*, 37(7):760–770, 2013.

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

H. T. Ng, S. M. Wu, Y. Wu, C. Hadiwinoto, and J. Tetreault. The CoNLL-2013 shared task on grammatical error correction. In H. T. Ng, J. Tetreault, S. M. Wu, Y. Wu, and C. Hadiwinoto, editors, *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–12, Sofia, Bulgaria, Aug. 2013. Association for Computational Linguistics. URL `https://aclanthology.org/W13-3601/`.

H. T. Ng, S. M. Wu, T. Briscoe, et al. The conll-2014 shared task on grammatical error correction. In *Proceedings of the CoNLL Shared Task*, pages 1–14, 2014.

D. Nicholls. The cambridge learner corpus: Error coding and analysis for lexicography and elt. In *Proceedings of the Corpus Linguistics 2003 Conference*, pages 572–581, Cambridge, 2003. Cambridge University Press.

OSCAR Project. Oscar - open super-large crawled aggregated corpus. `https://oscar-project.org/`, 2020. Accessed: 2025-06-20.

M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. In M. Walker, H. Ji, and A. Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1202. URL `https://aclanthology.org/N18-1202/`.

M. Polignano, P. Basile, M. De Gemmis, G. Semeraro, V. Basile, et al. Alberto: Italian bert language understanding model for nlp challenging tasks based on tweets. In *CEUR workshop proceedings*, volume 2481, pages 1–6. CEUR, 2019.

M. Rei and H. Yannakoudakis. Compositional sequence labeling models for error detection in learner writing. *arXiv preprint arXiv:1607.06153*, 2016.

M. Rei and H. Yannakoudakis. Auxiliary objectives for neural error detection models. In J. Tetreault, J. Burstein, C. Leacock, and H. Yannakoudakis, editors, *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 33–43, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-5004. URL `https://aclanthology.org/W17-5004/`.

M. Reznicek, A. Lüdeling, and H. Hirschmann. Competing target hypotheses in the falko corpus: A flexible multi-layer corpus architecture. In *Automatic treatment and analysis of learner corpus data*, pages 101–124. John Benjamins Publishing Company, 2013.

V. Sanh, L. Debut, J. Chaumond, and T. Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

A. Singh. A guide to softmax activation function. `https://www.singlestore.com/blog/a-guide-to-softmax-activation-function/`, 2023. Accessed: 2025-06-25.

Språkbanken. MultiGED: A Multilingual Grammatical Error Detection Benchmark. `https://spraakbanken.github.io/multiged-2023/`, 2023. Accessed: 2025-07-17.

J. Tiedemann. Parallel data, tools and interfaces in OPUS. In N. Calzolari, K. Choukri, T. Declerck, M. U. Doğan, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, editors, *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey, May 2012. European Language Resources Association (ELRA). URL https://aclanthology.org/L12-1246/.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

O. Vinogradova and O. Lyashevskaya. Review of practices of collecting and annotating texts in the learner corpus realec. In *International Conference on Text, Speech, and Dialogue*, pages 77–88. Springer, 2022.

E. Volodina, C. Bryant, A. Caines, O. De Clercq, J.-C. Frey, E. Ershova, A. Rosen, and O. Vinogradova. MultiGED-2023 shared task at NLP4CALL: Multilingual grammatical error detection. In D. Alfter, E. Volodina, T. François, A. Jönsson, and E. Rennes, editors, *Proceedings of the 12th Workshop on NLP for Computer Assisted Language Learning*, pages 1–16, Tórshavn, Faroe Islands, May 2023. LiU Electronic Press. URL https://aclanthology.org/2023.nlp4call-1.1/.

A. Yang, B. Yang, B. Hui, B. Zheng, B. Yu, C. Zhou, C. Li, C. Li, D. Liu, F. Huang, G. Dong, H. Wei, H. Lin, J. Tang, J. Wang, J. Yang, J. Tu, J. Zhang, J. Ma, J. Xu, J. Zhou, J. Bai, J. He, J. Lin, K. Dang, K. Lu, K. Chen, K. Yang, M. Li, M. Xue, N. Ni, P. Zhang, P. Wang, R. Peng, R. Men, R. Gao, R. Lin, S. Wang, S. Bai, S. Tan, T. Zhu, T. Li, T. Liu, W. Ge, X. Deng, X. Zhou, X. Ren, X. Zhang, X. Wei, X. Ren, Y. Fan, Y. Yao, Y. Zhang, Y. Wan, Y. Chu, Y. Liu, Z. Cui, Z. Zhang, and Z. Fan. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019.

H. Yannakoudakis, T. Briscoe, and B. Medlock. A new dataset and method for automatically grading esol texts. *ACL*, 2011.

D. Yarowsky. Decision lists for lexical ambiguity resolution: Application to accent restoration in Spanish and French. In *32nd Annual Meeting of the Association for Computational Linguistics*, pages 88–95, Las Cruces, New Mexico, USA, June 1994. Association for Computational Linguistics. doi: 10.3115/981732.981745. URL https://aclanthology.org/P94-1013/.

Z. Yuan, S. Taslimipoor, C. Davis, and C. Bryant. Multi-class grammatical error detection for correction: A tale of two systems. In M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8722–8736, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.687. URL https://aclanthology.org/2021.emnlp-main.687/.

T. Zesch and J. Haase. HOO 2012 shared task: UKP lab system description. In J. Tetreault, J. Burstein, and C. Leacock, editors, *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 302–306, Montréal, Canada, June 2012. Association for Computational Linguistics. URL https://aclanthology.org/W12-2036/.