ΜΕΤΑΠΤΥΧΙΑΚΟ ΔΙΠΛΩΜΑ ΕΙΔΙΚΕΥΣΗΣ
ΣΤΙΣ ΨΗΦΙΑΚΕΣ ΜΕΘΟΔΟΥΣ ΓΙΑ ΤΙΣ ΑΝΘΡΩΠΙΣΤΙΚΕΣ ΕΠΙΣΤΗΜΕΣ

Using Predictive Text for Grammatical Error Correction in Second Language Learning

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΚΑΤΕΡΙΝΑ ΚΟΡΡΕ

ΑΘΗΝΑ, ΝΟΕΜΒΡΙΟΣ 2020

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΣΧΟΛΗ ΕΠΙΣΤΗΜΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΤΗΣ ΠΛΗΡΟΦΟΡΙΑΣ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΜΕΤΑΠΤΥΧΙΑΚΟ ΔΙΠΛΩΜΑ ΕΙΔΙΚΕΥΣΗΣ
ΣΤΙΣ ΨΗΦΙΑΚΕΣ ΜΕΘΟΔΟΥΣ ΓΙΑ ΤΙΣ ΑΝΘΡΩΠΙΣΤΙΚΕΣ ΕΠΙΣΤΗΜΕΣ

Using Predictive Text for Grammatical Error Correction in Second Language Learning

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΚΑΤΕΡΙΝΑ ΚΟΡΡΕ

f36619910

Επίβλεψη: John Pavlopoulos

Κριτές: Panos Constantopoulos,

Ion Androutsopoulos

ΑΘΗΝΑ, ΝΟΕΜΒΡΙΟΣ 2020

Περίληψη

Η αυτόματη διόρθωση γραμματικών λαθών (Grammatical Error Correction, GEC) αφορά την αυτόματη διόρθωση διαφορετικών τύπων λαθών, όπως λάθη ορθογραφίας, στίξης, και γραμματικής. Ένα σύστημα GEC προϋποθέτει συνήθως την εισαγωγή μιας λανθασμένης πρότασης με σκοπό να τη μετατρέψει στη σωστή εκδοχή της. Υπάρχουν πολλές προσεγγίσεις για τη διόρθωση γραμματικών λαθών, από μοντέλα κανόνων (rule-based models) έως και νευρωνική μηχανική μετάφραση (neural machine translation). Υπάρχει παρόλα αυτά μια προσέγγιση η οποία δεν έχει μελετηθεί αρκετά: η γλωσσική μοντελοποίηση (language modeling) (Bryant and Briscoe, 2018), και πιο συγκεκριμένα η πρόβλεψη κειμένου. Τα γλωσσικά μοντέλα χρησιμοποιούνται ως επί το πλείστον για παραγωγή λόγου, πράγμα το οποίο μας κάνει να παραβλέπουμε την προοπτική τους να χρησιμοποιηθούν πιθανώς και ως ένα εργαλείο για την διόρθωση μιας λανθασμένης πρότασης ή για την αποφυγή λαθών κατά τη γραφή. Η μελέτη αυτή επικεντρώνεται σε αυτήν την προοπτική. Εκπαιδεύοντας και αξιολογώντας ένα στατιστικό γλωσσικό μοντέλο (SLM) και το νευρωνικό γλωσσικό μοντέλο αυτο-επιβλεπόμενης μάθησης GPT-2, εξετάζω τη ικανότητα των μοντέλων να προβλέπουν την διόρθωση σε προτάσεις οι οποίες περιέχουν γραμματικά λάθη. Τα αποτελέσματα έδειξαν πως τα γλωσσικά μοντέλα, όντας ρυθμισμένα να προβλέπουν τη στατιστικά πιο πιθανή διόρθωση, μπορούν να προβλέπουν περίπου το 15% των διορθώσεων. Το ποσοστό αυτό αυξάνεται περίπου στο 25% όταν το γλωσσικό μοντέλο προβλέπει τις 3 πιο πιθανές προβλέψεις. Για να ελέγξω την παιδαγωγική ικανότητα ενός γλωσσικού μοντέλου, διεξήγαγα ένα πείραμα με πραγματικούς μαθητές της Αγγλικής γλώσσας ως δεύτερη ξένη γλώσσα. Χρησιμοποιώντας το GPT-2 για να παράγει κείμενο το οποίο λειτουργεί ως πιθανή συνέχεια των προτάσεων των μαθητών, δημιούργησα ένα μικρό σώμα κειμένων των εκθέσεων των μαθητών και ανέλυσα τα λάθη τους μαζί με τις συχνότητες τους. Το πείραμα έδειξε πως τα γλωσσικά μοντέλα μπορούν όντως να βοηθήσουν τους μαθητές να γράψουν γραμματικά πιο σωστές εκθέσεις. Ωστόσο, αξίζει να σημειωθεί ότι το ποσοστό επιτυχίας εξαρτάται επίσης και από τα χαρακτηριστικά του/της εκάστοτε μαθητή/τριας.

Λέξεις κλειδιά: GEC, Πρόβλεψη Κειμένου, Εκμάθηση Δεύτερης Ξένης Γλώσσας

Abstract

Grammatical Error Correction (GEC) is the task of correcting different types of errors, such as spelling, punctuation, and grammatical errors, in written texts. A GEC system usually requires an input consisting of the erroneous sentence in order to transform it into the correct version of it. There are many approaches to GEC, from rule-based models to neural machine translation. However, there is one approach that has not been researched enough: language modeling (Bryant and Briscoe, 2018), and more specifically, predictive text. Language models are prominently used for language generation, a fact that makes us overlook the possibility of using them as a tool for predicting the correction, or the correct token in an erroneous sentence. This study focuses on this potential. By training and testing a statistical language model (SLM) and an autoregressive language model, GPT-2, I examined the potential of the models to predict the correct token in sentences that contain grammatical mistakes. The findings showed that language models can predict approximately 15% of the correct tokens with one greedy generated prediction. This percentage rises up to around 25% when the language model generates the top 3 predictions. To test the pedagogical capacity of a language model, I also experimented with real English as a second language (ESL) learners. By equipping state-of-the-art language model, GPT-2, to generate text that functions as potential continuation of the learners' sentences, I created a small corpus of the learners' writings and analyzed their errors along with their frequencies. The experiment showed that language models can actually help learners to write more grammatically correct essays, however, the rate of success also depends on the learners' individual characteristics.

Keywords: GEC, Predictive Text, SLA

Acknowledgements

I would like to thank my supervisor, John Pavlopoulos, for putting up with my countless emails and providing me with feedback and guidance throughout this thesis. Also, a big thanks to my students/participants of this study for their patience and willingness to help me, and without whom this thesis would have not been completed.

Contents

List of Tables

List of Figures

1 Introduction

Grammatical Error Correction (GEC) is the task of correcting different types of errors, such as spelling, punctuation, and grammatical errors, in written texts. The procedure of correcting a sentence that contains an error usually requires a system to use the erroneous sentence and transform it into the correct version of it. GEC applications can mainly be found in text processors, as well as in online writing services, such as Grammarly.[1] Such systems, apart from assisting the casual user in daily written tasks, can also assist second language (L2) learners to improve their writing skills in their target language.

This thesis is concerned with the pedagogical aspect of GEC in L2 learning, and in particular in learning and teaching the English language. Given that English is spoken by around 20% of the world's population as a foreign language, [2] there is an urgent need for new pedagogical methods that comply with a new technological framework, and which can offer adequate assistance both to learner and to educator. Due to the autonomy that GEC systems are rapidly acquiring, L2 learning applications will be able to both aid the learner's self-study and self-evaluation, and at the same time alleviate the educator's workload, such as correcting essays.

1.1 Purpose of the study

GEC undoubtedly has a lot of advantages to offer in L2 learning. Considering that there are multiple methods to create a GEC system (i.e., rules, classification, statistical machine translation, neural machine translation), the opportunity to create "tailored" applications suited to the learner and educator's needs seems all the more tangible. In addition, given that not all learners prefer a supervised pedagogical method, self-teaching through the assistance of GEC systems becomes even more effective. Learners will have the ability to detect and evaluate their own errors, as well as get proper feedback even with statistical representations. By using such technological tools, educators will be able to create curricula that are adapted to the class or the individual student's needs. For example, an educator knowing that the majority of the class does not comprehend the subject-verb agreement grammatical phenomenon will modify the curriculum with activities to practice the particular issue. Therefore, a data-driven pedagogical approach can have very positive effects in the classroom.

The increasing popularity of GEC as a Natural Language Processing, or NLP, topic is proven by the two most recent shared tasks, CoNLL-2014 (Ng et al., 2014) and BEA-2019 (Bryant, Felice, Andersen, & Briscoe, 2019). The two shared tasks involved system creators to create GEC systems that would correct the sentences of a multi-set of data of different groups of learners, and which consequently contained a great variety of grammatical errors. The two shared tasks not only presented state-of-the-art systems in the field of GEC, but also brought into the spotlight several weaknesses that still afflict modern systems, such as handling sentences that contain multiple errors.

[1] https://www.grammarly.com/

[2] https://www.britishcouncil.org/sites/default/files/english-effect-report-v2.pdf

1.2 Aims

Most of the systems that participated in the shared tasks used transformation methods to correct the erroneous sentences. This study does not deal with grammatical correction per se. In fact, it focuses on predicting an already given correction. The chosen NLP approach in this study is to use predictive text. First of all, the idea is to check whether language modeling can be successful at predicting the correction of an erroneous sentence. Usually, predictive keyboards or texts are evaluated disregarding the difficulty of the correct token to be predicted. Thus, a question remains answered: are language models more or less effective in predicting tokens that people are having difficulty with? The second aim of this study is to verify the hypothesis that predictive text can help in L2 learning. Specifically,

1. I attempt to evaluate two language models on the prediction of the correct token [3] in erroneous sentences. The language models used are:
    - A statistical N-gram language model.
    - State-of-the-art unsupervised neural language model GPT-2.
2. I experiment with the most successful language model by using it with real L2 learners. The steps of the experiment involve:
    - the learners writing 3 essays without the predictive text tool and 3 essays with the predictive text tool.
    - the essays are then corrected.
    - the original and corrected essays are then input to ERRANT, an automatic annotation toolkit (see Chapter 3.3), to reveal frequency patterns.
    - participants evaluate their experience with and without the predictive text tool.

1.3 Structure

Following this introduction, Chapter 2 discusses some important topics associated with this research. First, it presents the most popular GEC methods. Then, it introduces the two shared tasks, as well as underlining the most successful GEC systems that competed. Secondly, the importance of error type evaluation is discussed, followed by the introduction of the idea of predictive keyboard in L2 learning. Chapter 3 is an exploratory analysis of the data. Chapter 4 presents our methodology and results on correct token prediction of erroneous sentences via predictive keyboard, as well as mentioning vital attributes of the language models used. Chapter 5 concerns the experiment with real learners. After having presented an integrated picture of this study, Chapter 6 concludes this thesis by summarising the findings and discussing limitations, potential solutions and suggestions.

[3] The correct token is the correction of a mistake. For example, if we tokenized the sentence " This are a grammatical sentence", the token "are" is the error token and should be replaced with the correct token "is".

2 Background

This chapter presents the most recent work in GEC, while also discussing GEC in relation to L2 education by introducing some main concepts. The first section is an overview of the approaches used in GEC. The next two sections are concerned with the two shared tasks. The fourth section demonstrates the idea of Computer Assisted Language Learning (CALL), in conjunction with the importance of error type classification. Finally, the fifth section discusses the effects of predictive texts on human language and its potential in L2 learning.

2.1 Research approaches in GEC

Grammatical error correction can be defined as the automated feedback on a person's writing. While correcting the sentences, a GEC system has to make sure that the original meaning of the sentence stays the same once the sentence is transformed. Automating the task of grammatical error correction can benefit native speakers, people that learn a second language, and most of all people learning English as a second language (ESL) (Ailani, Dalvi, & Siddavatam, 2019). There are three main approaches for the creation of a GEC system so far:

- **Rule-based** approaches assure that the sentences follow specific manually coded grammar rules and that they match certain patterns. The grammar rules are usually based on Context Free Grammars, while Syntactical analysis is also integrated into the system. A parser is used to check if the Part of Speech (PoS) tags in the text comply with the given rules. Among the three approaches, the rule-based one is the easiest to implement. However, it is unable to cope with more complex errors, due to the fact that is impossible to define enough rule combinations (Ailani et al., 2019).

- **Classification-based** approaches are again error-type specific. Machine learning classifiers work with error-coded corpora and are built to correct the errors. The problem with classifiers is that each classifier can usually detect only one error type, neglecting any other error types in the sentence. For that reason, either a combination of classifiers or, even better, a classifier approach along with another approach (e.g. Statistical Machine Translation) is preferred (Ailani et al., 2019).

- **Machine Translation** approaches which are divided into **Statistical Machine Translation (SMT)** and **Neural Machine Translation (NMT)**. SMT uses parallel error-annotated data sets and can be used to solve any types of errors. An issue with SMT is that it can be dependent of the corpus size, while its efficiency might depend on contextual information (Ailani et al., 2019). NMT, on the other hand, uses an "Encoder-Decoder" mechanism. Specifically, the encoder reads the sentence and encodes it into a vector, while the decoder produces a translation. This is possible because the encoded vector can help predict the next word (Ailani et al., 2019).

As far as the evaluation of the systems is concerned, there are several evaluation metrics used to assess the performance of the systems. The most used ones are BLEU (Papineni,

Roukos, Ward, & Zhu, 2002), GLEU (Mutton, Dras, Wan, & Dale, 2007), and MaxMatch (M2) scorer (Dahlmeier & Ng, 2012). A recent addition to this list is the ERRANT scorer, which is a modification of the M2 scorer (Bryant, Felice, & Briscoe, 2017).

2.2 CoNLL-2014

The 18th Conference on Computational Natural Language Learning announced a shared task which focused on Grammatical Error Correction. The task is known as the CoNLL-2014 shared task (Ng et al., 2014). The aim of this task was to create an end-to-end application that detects and corrects all types of grammatical errors in a given essay. This was a challenge because, currently, not all grammatical error types can be properly detected and corrected by the given system, resulting in low performance. Therefore, teams participating in the shared task were called to evaluate algorithms and systems that would enable the automatic detection and correction of grammatical errors in written essays by learners of English as a second language. For the purposes of the study, two different corpora were used as the main data sets: the NUCLE Corpus and the NUS Corpus of Learner English, accounting a total of 1,312 sentences. From the 45 participating teams, the 13 submitted their output systems. The models were evaluated with the M2 scorer (Dahlmeier and Ng, 2012) which estimates a span-based Fβ-score, where β here is set to 0.5 to weigh precision twice as recall. There was a great variation in the scores, with the most successful team achieving an $F_{0.5}$ score of 37.33% with alternative answers and of 45.57% without alternative answers. The approach of the most successful team, from the university of Cambridge, was to develop a pipeline methodology which involved an initial rule-based approach, language model ranking, untuned SMT, then again language model ranking and, finally, type filtering (Felice, Yuan, Andersen, Yannakoudakis, & Kochmar, 2014). Most of the teams, also used external resources for corpora.

2.3 BEA-2019/2020

Successor to the CoNLL-2014 shared task, the Building Educational Applications shared task of 2019, also known as BEA-2019 shared task (Bryant et al., 2019), provides a wider range of data, with the addition of a new dataset, the Write&Improve+LOCNESS corpus. Moreover, it introduces the concept of tracks. There are three tracks that determine the amount of data that is available to the participant: the restricted track, in which the participants can use only the official datasets as annotated training data; the unrestricted track, in which participants can use all sets and any other resources to build systems; and the low-resource track, in which participants are allowed to use only the W&I+LOCNESS corpus.

The completion of the task offered both new insights, while it also underlined some already existing problems. As expected, the systems scored higher with the new dataset. However, some systems were more effective depending on the CEFR level while all systems showed weakness concerning content word errors. More specifically, the best results were acquired with the system of GECToR (Omelianchuk, Atrasevych, Chernodub, & Skurzhanskyi, 2020). By "using a sequence tagging approach, an encoder from a pretrained Transformer, custom transformations and 3-stage training" (Omelianchuk et al.,

Table 1

CoNLL-2014 results

Team ID	Precision	Recall	$F_{0.5}$
CAMB	39.71	30.10	37.33
CUUI	41.78	24.88	36.79
AMU	41.62	21.40	35.01
POST	34.51	21.73	30.88
NTHU	35.08	18.85	29.92
RAC	33.14	14.99	26.68
UMC	31.27	14.46	25.37
PKU*	32.21	13.65	25.32
NARA	21.57	29.38	22.78
SJTU	30.11	5.10	15.19
UFC*	70.00	1.72	7.84
IPN*	11.28	2.85	7.09
IITB*	30.77	1.39	5.90

Team ID	Precision	Recall	$F_{0.5}$
CUUI	52.44	29.89	45.57
CAMB	46.70	34.30	43.55
AMU	45.68	23.78	38.58
POST	41.28	25.59	36.77
UMC	43.17	19.72	34.88
NTHU	38.34	21.12	32.97
PKU*	36.64	15.96	29.10
RAC	35.63	16.73	29.06
NARA	23.83	31.95	25.11
SJTU	32.95	5.95	17.28
UFC*	72.00	1.90	8.60
IPN*	11.66	3.17	7.59
IITB*	34.07	1.66	6.94

Note. Tables taken from Ng et al. (2014). The left table demonstrates the scores (in %) without alternative answers, while the right demonstrates the scores with alternative answers. The asterisk next to certain team names indicates that they submitted their system output after the deadline.

Table 2

BEA-2019 results

Restricted

Group	Rank	Teams	TP	FP	FN	P	R	$F_{0.5}$
1	1	UEDIN-MS	3127	1199	**2074**	72.28	60.12	**69.47**
	2	Kakao&Brain	2709	894	2510	**75.19**	51.91	69.00
2	3	LAIX	2618	960	2671	73.17	49.50	66.78
	4	CAMB-CLED	2924	1224	2386	70.49	55.07	66.75
	5	Shuyao	2926	1244	2357	70.17	55.39	66.61
	6	YDGEC	2815	1205	2487	70.02	53.09	65.83
3	7	ML@IITB	**3678**	1920	2340	65.70	**61.12**	64.73
	8	CAMB-CUED	2929	1459	2502	66.75	53.93	63.72
4	9	AIP-Tohoku	1972	902	2705	68.62	42.16	60.97
	10	UFAL	1941	942	2867	67.33	40.37	59.39
	11	CVTE-NLP	1739	811	2744	68.20	38.79	59.22
5	12	BLCU	2554	1646	2432	60.81	51.22	58.62
6	13	IBM	1819	1044	3047	63.53	37.38	55.74
7	14	TMU	2720	2325	2546	53.91	51.65	53.45
	15	qiuwenbo	1428	854	2968	62.58	32.48	52.80
8	16	NLG-NTU	1833	1873	2939	49.46	38.41	46.77
	17	CAI	2002	2168	2759	48.01	42.05	46.69
	18	PKU	1401	1265	2955	52.55	32.16	46.64
9	19	SolomonLab	1760	2161	2678	44.89	39.66	43.73
10	20	Buffalo	604	**350**	3311	63.31	15.43	39.06
11	21	Ramaiah	829	7656	3516	9.77	19.08	10.83

Note. Table from Bryant et al. (2019). Here only the results of the restricted track are presented. The results of the other two tracks can be found in the appendix.

2020), this system managed better and faster results than a Transformer-based seq2seq system. Combining grammatical error correction systems was also very successful, such as in Kantor et al. (2019). The goal of that approach was to automatically detect the strength of a specific system or the combination of many systems at the same time to improve precision. In addition, the research team attempted to outperform RNNs, analyse BERT, and, finally, present a spellchecker that was specifically created for the task, and which aspired to be better than the existing spellcheckers. By combining several systems, or even combining one system with itself, they managed to have more accurate results and outperform most single systems. This combination of systems also rendered the detection and learning of certain subtle distinctions of grammatical error types possible.

Kiyono, Suzuki, Mita, Mizumoto, and Inui (2019) concentrated on the hypothesis that grammatical error correction models can be improved by implementing pseudo data in the training process. They used three different methods to generate ungrammatical sentences: noisy backtranslation, sample backtranslation, and direct noise. After experimenting with and incorporating the pseudo data into different GEC systems, they achieved state-of-the-art results both in on the ConLL-2014 and the BEA-2019 test sets.

Another problem that was tackled through this shared task was the tendency of the data used as input in a GEC model to be considerably differently distributed from that used for pre-training a masked language model (MLM) (Kaneko, Mita, Kiyono, Suzuki, & Inui, 2020). By embedding a pre-trained MLM (BERT) into their own GEC model and by fine-tuning the MLM with GEC corpora from the task to add furthermore features in the model, Kaneko et al. (2020) demonstrate how a fusion with MLMs can benefit a GEC model.

2.4 Error Types and CALL

In this section, I will attempt to present the most common grammatical error types produced by non-native English speakers during and after second language acquisition, as well as highlight the importance of the classification of those errors in Computer Assisted Language Learning, a.k.a CALL. It has to be noted, that error analysis in second language acquisition is language specific, meaning that the results of each study concern native speakers of different languages whose target language is English. It is also important to acknowledge that most of error analysis studies, so far, are concerned with Asian native speakers because of the major differences in the structure between Asian languages and English, e.g the SVO[4] vs SOV structure of the sentence. Despite those interlinguistic obstacles, error analysis could not only enhance ESL learners' writing, but it could also enable teachers to determine the current level of the learners in the language learning process, and researchers to determine how language is learned and structured. By automating error analysis, second language learning can become more efficient.

To begin with, grammatical errors can be due to various reasons, such as lack of adequate knowledge of grammar, interference of mother tongue, as well as lack of sufficient practice (Kraichoke, 2017). On a first level, those errors can be categorised on the basis of two different aspects: whether they are influenced by the learner's mother tongue or not.

[4]Subject-Verb-Object

If so, we are talking about "Interlingual or Transfer Errors" (Selinker, 1972). Otherwise, the errors fall into the second category, that of "Intralingual or Developmental Errors" (Richards & Richards, 1974), which are errors that do not reflect the learner's structure of mother tongue, but they are caused by the learner's inadequate exposure to the target language (Wu & Garza, 2014). Wu and Garza's study focused on the error analysis in the compositions of Chinese students of English as a foreign language (EFL), and their findings are presented in Table 3 along with some examples for further comprehension.

Table 3

Main error categorization in the compositions of Chinese EFL students

Number	Error Type	Example	Correction
1	S-V agreement	. . . **is** there any more documents	are
2	Fragment	From Quito, Ecuador	Insert 'I am'
3	Sentence Structure	I will not need a full scholarship but a partial one as I intend to pay	, but a partial one,
4	Singular/Plural	I'd like to make a few **inquire**	inquiries
5	Verb Omission	Also, any scholarships available	insert 'are'
6	Subject Omission	want to know something	insert 'I'
7	Coordination	The bus driver is a careless person, and he was pulled over by the police several times last week.	The bus driver, who is a careless person , was pulled over by the police several times last week
8	Relative Clause	They went to the same restaurant that Mark had been to it	omit **it**
9	Verb Tense	I am looking forward to hear from you	hearing

All of the error categories can be expanded into subcategories, for example faulty coordination or construction of a relative clause, can occur with the use of a different conjunction or relative pronoun accordingly. Another example is comparison, where there are many ways to compare two or more entities, apart from using the comparative and superlative form, by using constructions such as *as...as*. A possible ungrammatical sentence using this structure would be "She run as **faster** as she could" with *fast* being the correction of the faulty sentence.

As mentioned above, error analysis can prove to be very useful both for students, when it comes to learning from their own or others' mistakes, and teachers who can in this way underline what students should avoid. Moreover, it enables them to get a better grasp of what goes on in the students' mind and helps them create a more efficient curriculum.

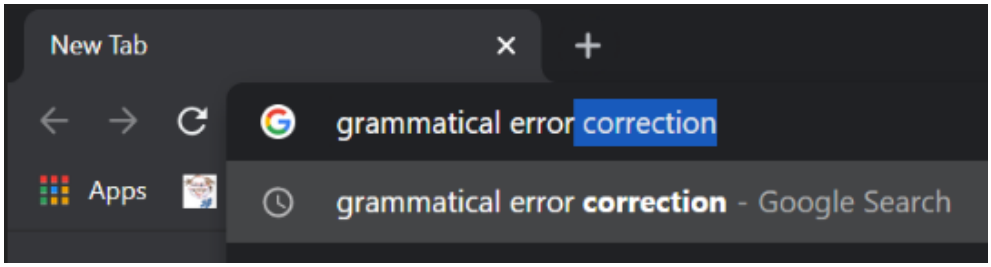
Aside from the benefits in the classroom, grammatical error analysis has also slowly started to step into CALL. Until very recently, the staple technique to NLG (Natural Language Generation) for language learning purposes, was to train models on large bodies of correct English (Lee & Seneff, 2008). Although this technique has proven to be effective, a more recent one seems to take into account more parameters when it comes to non-native speakers. This new technique involves relying on two kinds of corpora: a source corpus from non-native texts, and a target corpus, which, in reality, is a corrected version of the source corpus. It has to be noted that because these corpora are very limited, errors can be inserted into the well-formed text, according to patterns that have been observed in the non-native corpus (Lee & Seneff, 2008). The advantages of this technique is that: 1. it can help in the optimization of error frequency in the training dataset, 2. it can improve the

authenticity of the data-set, 3. this kind of simulated data can be turned into exercised of the cloze format. Meurers (2012, p. 6) summarises the benefits of the analysis of learner corpora by stating that "the annotation of learner corpora essentially provides an index to learner language properties in support of the goal of advancing our understanding of acquisition in SLA[5] research and to develop instructional methods and materials". However, he also makes a very important distinction stating that "[t]he goal of writer's aids is to support the second language user in writing a functional, well-formed text, not to support them in acquiring the language as is the goal of an ILTS[6]" (Meurers, 2012, p. 3). Taking this into consideration, it becomes obvious that CALL methods cannot actually function as a "cure-all" for language learning. It provides a significant amount of tools for learning, but it is up to the learners how to use them to simply facilitate them in their tasks in their target language.

2.5 Predictive Text

Predictive keyboard is omnipresent in all of our digital devices, from computers to tablets and mobile phones. The speed and convenience that it provides during typing has now made it an integral part of any writing tasks. Despite the common misconception that such conveniences might impair one's language abilities, a misconception mainly based on the fact that writing tools like predictive keyboard might reduce the activity of the brain, new studies suggest otherwise.

Figure 1: Predictive text example in internet browser



Predictive text can not only help in faster typing but it can also, in conjunction with auto-correct, improve the users spelling and grammatical skills. Waldron, Wood, and Kemp (2017) observed that predictive text can influence the quality of errors primary school students made, as well as that university students made significantly fewer grammatical mistakes when using predictive text. Cohort effects and age, however, can influence the capacity of such tools. Y. Kalman, Kavé, and Umanski (2015) conducted an experiment focusing on the use of predictive keyboard in younger and older age groups, in terms of speed and accuracy. As expected, there were differences in the scores of the two groups with the younger group typing faster and with a greater variation of keys, while the older groups typed more slowly and with less variation of keys. These findings suggest that a better understanding of the variables can contribute to personalized

[5]Second Language Acquisition
[6]Intelligent Language Tutoring Systems

Human-Computer Interaction (HCI) designs (Gajos, Hurst, & Findlater, 2012). Moreover, given that language and cognitive ability are interrelated, such studies can provide information on markers for cognitive decline or even injury (Y. M. Kalman, Geraghty, Thompson, & Gergle, 2012).

The benefits of predictive text in relation to cognitive skills can be traced from the 1990s, when PAL, a predictive computer program, was used in a classroom environment consisting of children with learning difficulties (Newell, Booth, & Beattie, 2006). PAL works differently from a usual predictive keyboard. "It exploits the redundancy in natural language to reduce the number of character entries necessary to produce a piece of text" (Newell et al., 2006, p.23). In this way PAL manages to offer some predictions that function as the continuation of the user's sentence, reducing thus the typing time. In Newell et al. (2006)'s study, 8 out of 9 study cases showed very positive results. PAL helped produce higher quality writings with reduced spelling errors, while it also enhanced the children's confidence and motivation.

In terms of the quality of writing with a predictive text, Arnold, Chauncey, and Gajos (2020) underline that aside from speed and accuracy, it is mandatory that we evaluate the effect intelligent text has on the content written. More specifically, their findings show that when the users were presented with the predicted options, they tended to write predictable sentences with fewer words. The two studies bring the two sides of the coin to the limelight, and address the potential benefits and shortcomings of predictive text. It is obvious then that there are certain effects of the predictive text on the native language users. The next question that needs to be addressed is what the effects of the predictive text in second language learners are. A very interesting hypothesis is that L2 learners will be influenced differently from native speakers, if we take into account that the former group does not anticipate information during processing to the same degree the latter group does (Kaan, 2014). How the second language learners react is one of the objectives of this thesis.

3 Exploratory Data Analysis

The intention of this chapter is to delve further into the data sets used for the purposes of this study, as well as bring to light and explore several problematic areas. As mentioned in the introduction, our data sets comprise the corpora used in the BEA-2019 shared task (Bryant et al., 2019), and therefore in this section I will also discuss the data analysis conducted for the shared task. Although the analysis in Bryant et al. (2019) provides a detailed description of the data set, it does not address the shortcomings in a thorough manner. This chapter focuses on two aspects. It presents a detailed description of the data by combining the analysis conducted for the BEA-2019 shared task paper with this analysis in my study; then it explores the shortcomings of the data, ultimately providing suggestions for improvement.

3.1 Datasets and format

To begin with, the data is a compilation of both native and second language learner corpora. In particular, it consists of:

- **the Cambridge English Write & Improve corpus**, which consists of 3,600 submitted texts, in the form of essays, letters, stories etc.(Yannakoudakis, Øistein E Andersen, Geranpayeh, Briscoe, & Nicholls, 2018), to the Write and Improve online platform.[7] The submissions were made by non-native English students. The annotators correct the texts manually. Any submission that met any of the following conditions was eliminated:
  - The text contained fewer than 33 words.
  - More than 1.5% of all characters in the text were non-ASCII.[8]
  - More than 60% of all non-empty lines were both shorter than 150 characters and did not end with punctuation (Bryant et al., 2019).
- **the LOCNESS corpus**,[9] which is a collection of 400 native English speakers undergraduate essays (Granger, 1998). Essays longer than 550 words were removed. In addition, because there were not enough texts for creating a training data set, LOCNESS was used only as the development and test set in the BEA-2019 shared task. (Bryant et al., 2019).
- **the National University of Singapore Corpus of Learner English (NUCLE)**, which contains 1,400 essays by Asian undergraduate students of the National University of Singapore (Dahlmeier, Ng, & Wu, 2013).
- **the First Certificate in English corpus (FCE)**, which consists of 1,244 answers to FCE writing task (Yannakoudakis, Briscoe, & Medlock, 2011).

[7] https://writeandimprove.com/
[8] American Standard Code for Information Interchange
[9] https://uclouvain.be/en/research-institutes/ilc/cecl/locness.html

- **Lang-8 Corpus of Learner English**, which is derived from the Lang-8 language learning website,[10] where users are able to correct each other's grammar (Tajiri, Komachi, & Matsumoto, 2012).

For the purposes of BEA-2019, all data sets were re-annotated and standardized with ERRANT, an automatic annotation toolkit which we will discuss in the next section. The output of ERRANT is the data in M2 format as seen in Table 4.

Table 4

ERRANT *M2 format example.*

```
S This are gramamtical sentence.
A 1 2|||R:VERB:SVA|||is|||REQUIRED|||-NONE-|||0
A 2 2|||M:DET|||a|||REQUIRED|||-NONE-|||0
A 2 3|||R:SPELL|||grammatical|||REQUIRED|||-NONE-|||0
A -1 -1|||noop|||-NONE-|||REQUIRED|||-NONE-|||1
```

Note. The line starting with S is the original sentence, while the ones starting with A are the edit annotations. The edits contain the start and end token offsets, the error type, the correction, a flag indicating whether the edit is required or optional, a comment field, and a unique annotator ID. A 'noop' edit indicates that no changes were made to the sentence. All of the annotated error types can be found in the appendix.

3.2 Data Analysis

On a first level, errors in the form of word edits are categorised in terms of whether they need (R) Replacement, a word is (M) Missing or is (U) Unnecessary. The following table, which a is modification of the data analysis in the BEA-2019 paper, presents the frequencies of these types across all data sets.

Table 5

Frequencies in terms of the distribution of Missing (M) Replacement (R) and Unnecessary (U) word edits.

(R) Replacement	M (Missing)	U (Unnecessary)	UNK (Unknown)
60-65%	20-25%	10-15%	2-3%

Note. M edits are lower in the FCE and NUCLE corpora, while U edits reach approx. 20% in NUCLE.

Bryant et al. (2019) claim that a possible explanation for the variances in the frequencies in terms of the distribution of Missing (M) Replacement (R) and Unnecessary (U)

[10] https://lang-8.com/

Table 6

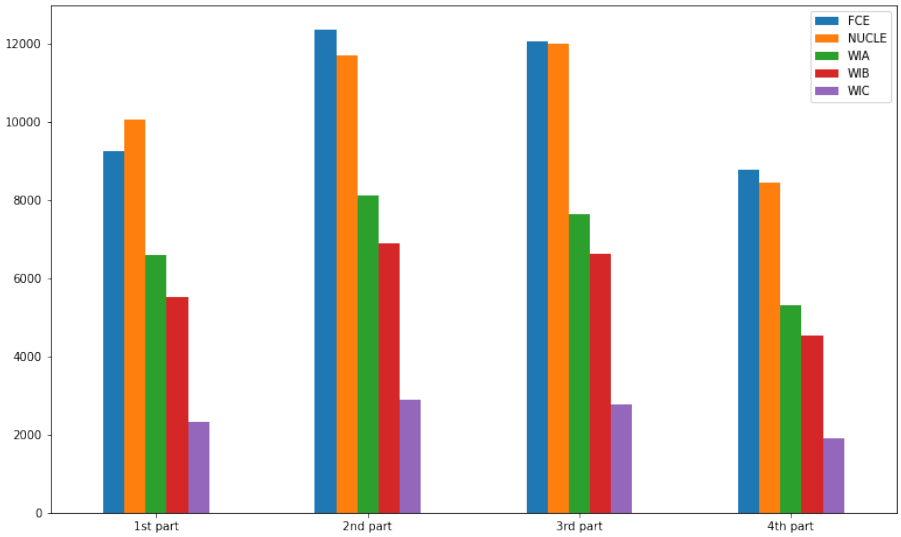
Basic corpus statistics.

Dataset	Total sentences	Erroneous sentences
FCE	28350	18045
Lang-8	1037561	497703
NUCLE	21835	21835
W&I(A)	10493	8330
W&I(B)	13032	9243
W&I(C)	10783	5472
Total	1122054	560628

Note. Almost 50% of the sentences contained errors. However, it must be noted that the NUCLE data set did not include any sentences that did not need any correction, thus only contributing to increasing the percentage of the erroneous sentences.

word edits (Table 5) could be attributed to different individual factors. One example is the fact that Asian learners find the use of determiners (the, this, that, etc.) difficult, hence the higher frequencies of DET errors in the NUCLE corpus. In addition, no UKN (unknown) errors existed in the Lang-8 corpus because they were differently annotated from the rest.

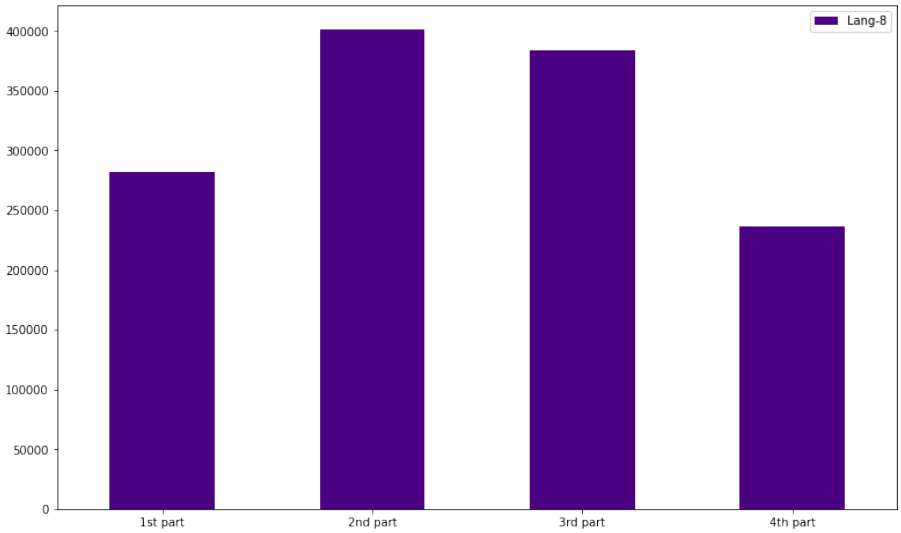
Figure 2: Relative location of errors in FCE, NUCLE, and W&I.



Note. Lang-8 error relative locations are presented in a separate figure as it accounts for a greater number of sentences which could not be adequately represented in the accumulative figure above.

Another aspect of the data that I tried to explore is error location. The initial data

Figure 3: Relative location of errors in Lang-8.



Note. Lang-8 error relative locations behave similarly as the rest of the data sets.

point at the error location with the starting and ending offsets, which basically reveal the range of the error in the tokenized sentence. Using these offsets, I calculated the relative location of the error, by dividing one of the offsets with the total count of the word tokens of each sentence. Then, I used intervals to determine whether the error of each sentence occurs at the beginning (0.00-0.25), middle-left (0.25-0.50), middle-right (0.50-0.75), or end (0.75-1.00) of the sentence. The results are presented in Figures 2 and 3. Lang-8 was plotted in a different bar graph because of the huge volume of the sentences that could not be presented together with the rest of the data sets.

Table 7

Example of relative location calculation.

Sentence	Offsets	Total Word Count	Relative Location
I was very disappointed after this show.	5 6	7	0.71

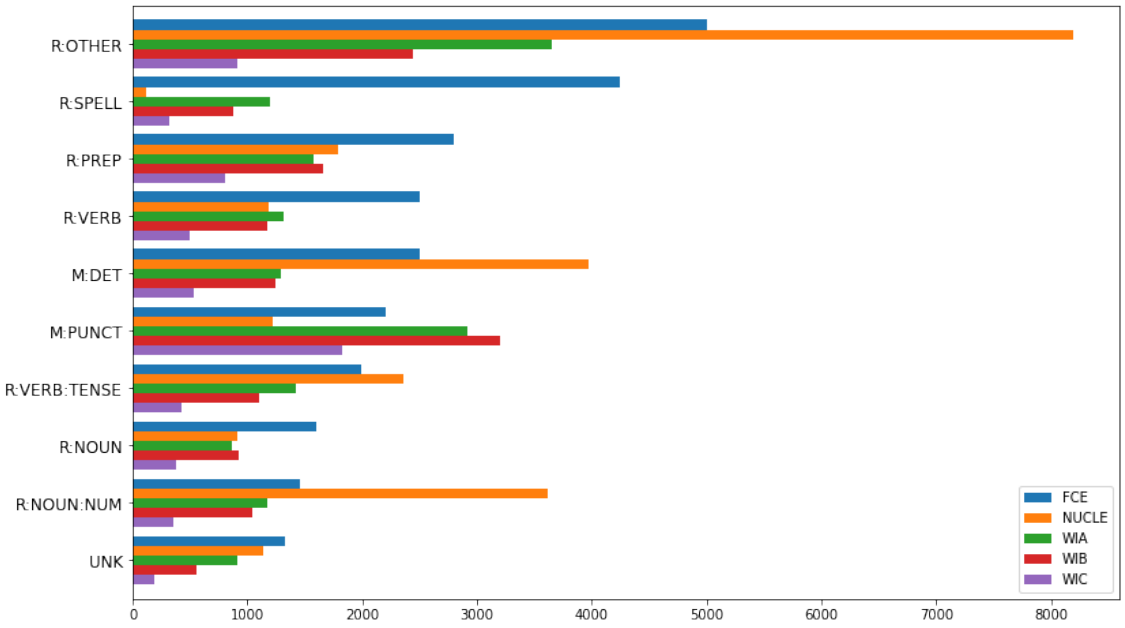
Note. In this example, the offsets indicate that the error lies between tokens 5 and 6. Therefore, we can estimate the relative location by dividing the starting offset (or the ending offset) with the total word count (5/7=0.71). So, if the relative location is 0.71 like in the example, then the error can be placed in the 3rd interval, in other words, in the third fourth of the sentence.

All data sets seem to behave in the same manner. Most mistakes are observed in the two middle parts of the sentences, while the two extreme parts of the sentences tend to have fewer mistakes. It has to be noted that when we use the ending offset to calculate the relative location, the most frequent locations of the errors are the the 3rd and 4th parts of the sentence.

To check if this distribution is random, I conducted a simple statistical experiment by calculating the p-value for the respective location. We consider our null hypothesis (H_0) that the distribution of errors is random. To test this, I generated a sample of a 1000 sentences and for each sentence I estimated whether the mistake falls into the two middle parts or not. I repeated the experiment 1000 times using a for loop. The result was a p-value>0.05, confirming our null hypothesis, and therefore proving that the tendency of the mistakes to appear in the middle of the sentence is not statistically significant.

Apart from error location, I considered error types a vital aspect of the data, especially when it comes to ESL. As mentioned in the Chapter 2, error types can provide great insights into the learners' learning pace and error patterns, and, therefore the study of error types can equip educators with information for a curriculum that fits each learner individually. To obtain a better idea on the error types of this data set, I worked out some frequency ratios. The frequency of each error type per data-set (upon pre-processing, see Section 4.1) in shown in the following figures. Figure 4 demonstrates the frequencies of the top 10 error types in all data sets apart from Lang-8 (Figure 5). The frequencies of all error types in each individual data set can be found in the appendix.

Figure 4: Error type frequencies.

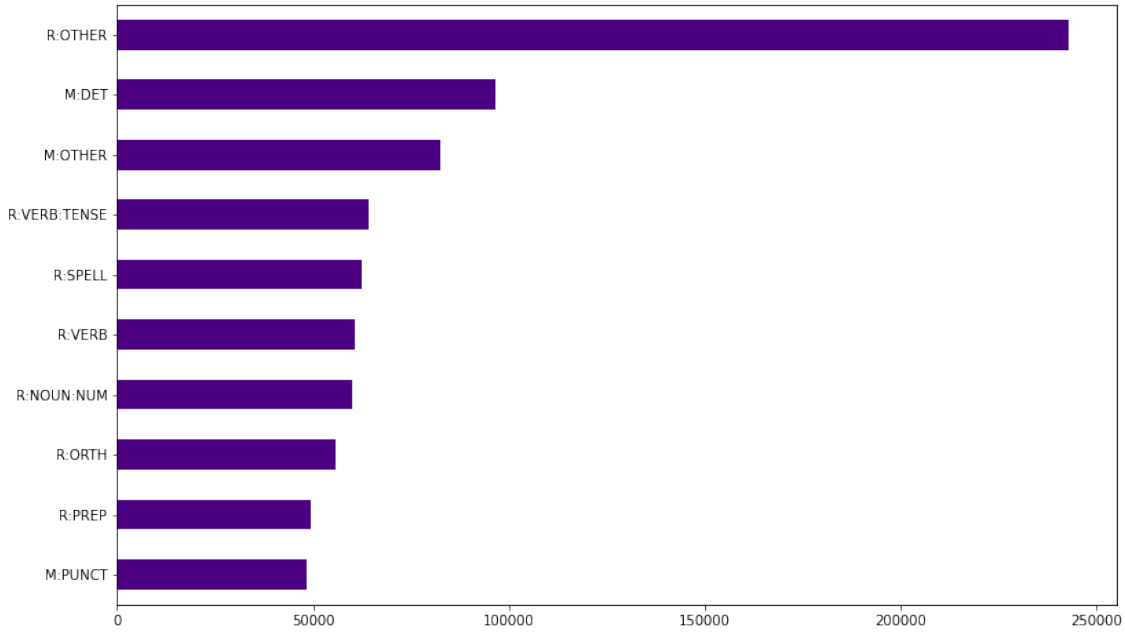


Note. 10 most frequent error types among data sets are presented in this figure. R:OTHER is the most frequent error type in the FCE, NUCLE, and WIA. M:PUNCT is the most frequent error type in WIB and WIC. Lang-8 error type frequencies are presented in a separate figure.

In both Figures, we can see that R:OTHER error type occupies first or second positions in terms of frequency. Bryant et al. (2017, p. 795) define this category as "[e]rrors that do not fall into any other category (e.g. paraphrasing)". After a qualitative analysis,

it became apparent that this particular error category contained errors that could have been included in other, more concrete categories, in which the error type would be more adequately described. One justification for this, according to Bryant et al. (2019, p. 55), is that "certain edits are longer and noisier...and do not fit into a more discriminate ERRANT category". This issue provides material for further investigation, and it will be further discussed in the next section.

Figure 5: Lang-8 error type frequencies.



Note. Lang-8 error type frequencies are presented in a separate figure as it accounts for a greater number of sentences which could not be adequately represented in the accumulative figure above.

Keeping in mind that each data set comes from a different demographic of learners, it is expected that the frequencies of error types vary among data sets. For example, a great portion of error types is assigned as M:PUNCT, namely missing punctuation. More specifically, and as it is also mentioned in Bryant et al. (2019), in NUCLE punctuation errors occur at a percentage of 5% while in W&I is rises to 20% if we add the percentages of each individual subdata-set.[11] This is also visualized in this study, where in two out of three Write & Improve subdata-sets, the most frequent error type is M:PUNCT. This difference might be due to the fact that W&I has a wider range of learners. Another observation made by Bryant et al. was that NOUN:NUM errors, that is noun number errors, occur twice the times in NUCLE compared to the rest of the corpora. Similarly to subject-verb agreement (SVA) errors, NOUN:NUM was among the five targeted error categories in ConLL-2013 shared task, hence the higher proportion.

[11] Write and Improve is divided into three subdata-sets: A, B, C

To conclude, this data analysis provided some great insights into the data that will be proven useful in the next sections. As anticipated, there are variations, both in terms of Missing, Replacement, and Unnecessary word tokens, as well as error type frequencies. These variations can be attributed to the fact that each data set comes from learners with different learning profiles. However, regarding the location of the errors, the analysis of the data suggests that there is a tendency of errors to appear in the middle of the sentence. Finally, OTHER error type was the most frequent error type, posing, therefore the question of whether ERRANT can be accurate enough in regards to error type classification. The next section focuses on this particular question.

3.3 ERRANT[12]

The fact that the majority of the error types in most corpora fell into the OTHER category poses the question of what kind of errors are included in this category, and, consequently, whether they are correctly assigned. For this reason, we conducted a qualitative examination of a sample of the data set to answer the two aforementioned questions. The results showed that indeed, many of the errors assigned with the OTHER label, could be placed into other categories. This chapter suggests an ERRANT improvement, by observing this major shortcoming that currently applies and suggesting the way for it to be addressed. More specifically,

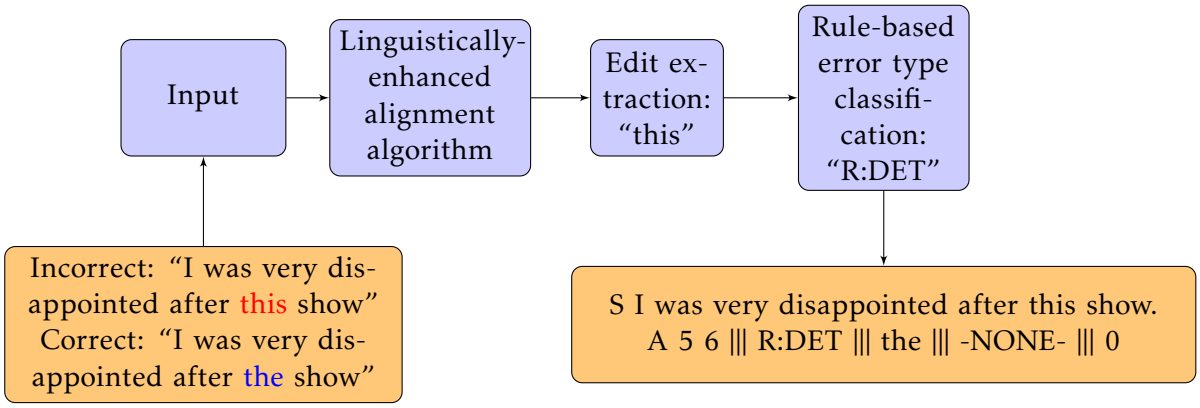
- We demonstrate a number of false or ambiguous classifications, using a sample of the FCE data set (Yannakoudakis et al., 2011). Although the error classifier has been evaluated to some degree (Bryant et al., 2017), we firmly believe that more investigation is needed.
- We suggest re-classifications of the detected faulty items. In specific, we estimate that 39% of what has been classified as error type OTHER (the most frequent type), should have been classified to other, known error types (e.g., R:VERB).
- We publicly release our detected false classifications and our suggested re-classifications, in order to initiate a collaborative, ongoing correction process of improving the FCE dataset, which we will use for a future robust training of machine learning classifiers. In this way, we believe that any ERRANT evaluation scorers can be improved (e.g., ERRANT was employed by the most recent Grammatical Error Correction shared task: BEA-2019 (Bryant et al., 2019)).

We will first present our approach to analysing the mis-classification problem. Then we will discuss our observations on mis-classification frequencies and patterns, along with possible implications in GEC.

Grammatical Error Correction (GEC) is the task of correcting different types of errors in written texts, usually by taking erroneous sentences as input and transforming

[12]This chapter is co-authored with John Pavlopoulos and it is accepted to the The 4th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (https://sighum.wordpress.com/events/latech-clfl-2020/).

Figure 6: ERRANT system demonstration.



Note. After the input, the linguistically enhanced-algorithm aligns the two parallel sentences by making sure that items with similar linguistic properties are aligned. R:DET means that the determiner 'this' needs to be replaced with the determiner 'the'.

them into correct ones. This can be achieved with a variety or even combination of techniques, such as language modeling (Bryant & Briscoe, 2018), statistical machine translation (Katsumata & Komachi, 2019), and neural machine translation (Grundkiewicz & Junczys-Dowmunt, 2018). An important step that is usually taken in these techniques is error tagging, namely "when all errors in the corpus have been annotated with the help of a standardized system of error tags" (Granger, 2003). Error tagging (or error classification) is of utmost importance as it contributes to sentence transformations in a GEC system, when the error is mapped to the correction through special tags, such as in (Omelianchuk et al., 2020). The most popular error tagger to date is the grammatical ERRor ANnotation Toolkit (ERRANT), which automatically extracts and categorizes errors from parallel original and corrected texts (Bryant et al., 2017). By employing a rule-based classifier, ERRANT is able to expand to other languages, such as German (Boyd, 2018), Spanish (Davidson et al., 2020) and Czech (Náplava & Straka, 2019). This fact makes it particularly important for second language (L2) learning, where it can provide automatic evaluation of GEC systems in several languages (Boyd, 2018; Náplava & Straka, 2019; Davidson et al., 2020).

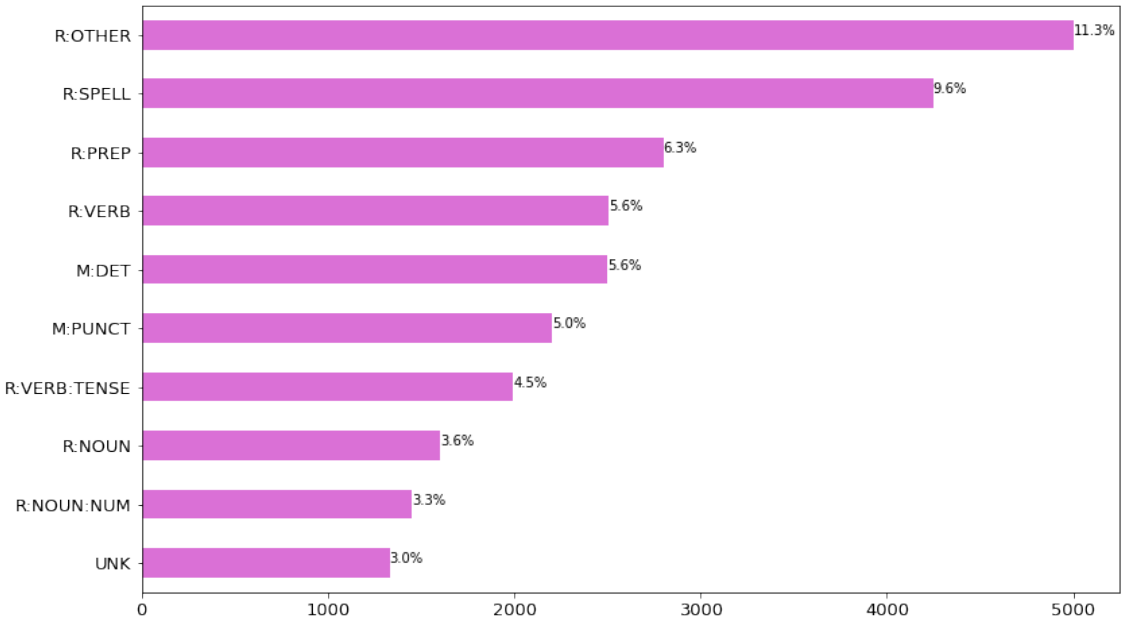
3.3.1 Approach

For the purposes of this study, we are only concerned with the FCE corpus (Yannakoudakis et al., 2011). A more extended study would focus on other datasets, as well. We used the FCE data file from the BEA-2019 shared task which was in M2 format and included all the extracted edits, error types and corrections. The exploratory data analysis showed that the most frequent error type was R:OTHER (see Figures 4 and 5), meaning that something in the sentence needs to be replaced with something else that does not fit into a certain category. Also, there were errors of type M:OTHER and U:OTHER, i.e. something is missing and something is unnecessary, respectively. We focused our analysis only on sentences

containing the most frequent error type, namely OTHER.

We sampled the first 100 sentences from the FCE corpus that contain OTHER type errors (incl. M:OTHER and U:OTHER) and we manually re-labeled each of them. All of our re-classifications are publicly released as an XLSX file,[13] along with the original uncorrected sentences, the starting and ending offsets, the suggested correction, and any comments.

Figure 7: Most frequent error types in the FCE dataset, where R:OTHER type errors comprise the majority of error types



Note. OTHER is the most common error type by far.

Table 8

Three main error categories.

Code	Meaning	Description	Example
ADJ:FORM	Adjective Form	Comparative/Superlative adjective errors	more easy (easier)
ORTH	Orthography	Case and/or whitespace errors	Bestfriend (instead of best friend)
VERB:INFL	Verb inflection	Missaplication of tense morphology	getted (got), fliped (flipped)

Note. Randomly selected out of the 25 presented in Bryant et al. (2017). All of the categories can be found in the appendix.

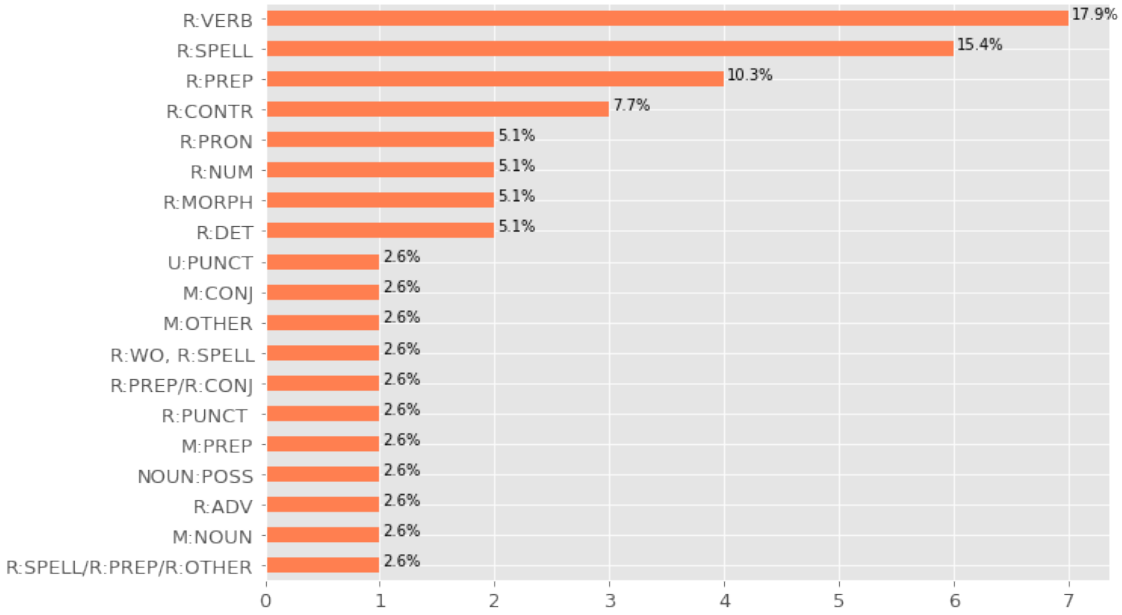
3.3.2 Results & Discussion

According to our re-classification, 39% of the errors could have been placed in other categories (i.e., 39 errors out of the sample of 100 sentences with one error each). Given

[13] https://github.com/katkorre/ERRANT-reclassification

that OTHER is the most frequent error type, a large number of sentences of the FCE corpus could potentially be re-classified to other categories. If this percentage applied to the whole FCE dataset, this would mean that 2724 out of the 6984 OTHER errors, are currently mistakenly tagged as OTHER.

Figure 8: Error type frequencies (prev. tagged as OTHER).



Note. Most frequent error type is R:VERB followed by R:SPELL and R:PREP. Replacement errors are also more frequent than missing or unnecessary word edit errors.

The most frequent error type that was classified as OTHER was R:VERB, namely a word in the sentence has to be replaced with a verb. Spelling mistakes (R:SPELL) were also very common, accounting for about 15% of the sample. Preposition replacements (R:PREP) comprised about 10% of the sample. There were errors that were placed in other categories, as well, but as figure 8 shows, they account for smaller percentages.

A more qualitative demonstration is presented in Table 9. Examples 1, 3 and 4 in Table 9 contain preposition replacement errors. Example 1 and 4 are cases that possibly reflect a greater issue of ERRANT. In particular, ERRANT seems to find it easier to properly classify errors that belong to the same part of speech as their correction, possibly as a result of its linguistically-enhanced alignment figure, which aligns items that are similar linguistically (Bryant et al., 2017).

In the examples, the words 'because' and 'and' are conjunctions and need to be replaced with the prepositions 'for' and 'at' respectively. Therefore, we are dealing with different parts of speech. ERRANT ignores the option to classify the errors as R:PREP (our suggestion), and classifies them as R:OTHER instead. The specific mis-classification could be explained if we take into consideration the linguistically-enhanced alignment algo-

rithm, which aligns linguistically similar items (see Figure 6). Because conjunctions and prepositions are different POS, ERRANT fails to assign the correct error type.

This is not the case for example 3 where the wrong preposition is replaced with a correct preposition, yet ERRANT does not provide the correct classification again. ERRANT seems to be also neglecting grammatical rules which have possibly not been implemented during the creation of ERRANT (see Figure 6 for the annotation process). For example, in sentence 2, the original sentence contains a wrong determiner 'a' in 'a person' and needs to be substituted with 'one'. In this case, the cardinal number 'one' becomes the determiner, hence the suggested error classification R:DET. Example 5 clearly contains a spelling mistake, but has been overlooked by ERRANT and has been put in the R:OTHER category. The error in example 6 was re-classified from R:OTHER to R:VERB. A hypothesis for the initial mis-classification could be that 'put in' is a phrasal verb, and again the linguistically-enhanced alignment algorithm prevented the correct classification. The last example could be re-classified either as R:PRON or as R:SPELL. The inability of the tool to choose between the annotation could be the reason behind the mis-classificaion.

Table 9

Examples of the qualitative re-classification

No	FCE Sentence	Offsets	Correction	Old type	New type
1	On the other hand , the theatre restaurant was closed because unknown reasons.	10 11	for	R:OTHER	R:PREP
2	There was only a person who used to call her by this name .	3 4	one	R:OTHER	R:DET
3	I want to thak you for preparing such a good programme for us and especially for taking us to the river trip to Greenwich..	18 19	on	R:OTHER	R:PREP
4	It is in the Central Exhibition Hall and will start and ten o'clock and finish at five o'clock in the evening .	10 11	at	R:OTHER	R:PREP
5	What are you dong here, why are n't you at school ? '	3 4	doing	R:OTHER	R:SPELL
6	Also supermarket owners have put in a vast amount of money to find out the best way to place goods in order to get the most profit .	4 6	invested	R:OTHER	R:VERB
7	Your sincerely	0 1	Yours	R:OTHER	R:PRON/ R:SPELL

Note. Example FCE sentences that are tagged as OTHER (5th column), along with their token-based offsets (3rd column, also highlighted in red in the text) and corrections (4th column). The last column presents our suggested re-classification.

Issues like the aforementioned must not be ignored. A more robust categorization might possibly lead to a more accurate grammatical error detection and, consequently, more efficient grammatical error correction systems.

ERRANT was used in the two most recent Grammatical Error Correction shared tasks (CoNLL-2014, BEA-2019), where all system output was automatically annotated with

the scorer of the toolkit (Bryant et al., 2019). Then, the automatically inferred error type was used by the participants to evaluate their performance per type. What this means, however, is that the participants are now misjudging their systems. If we assume the existence of an oracle system that always detects correctly the error type in a (FCE) sentence, then approx. 20% of the correctly detected R:VERB errors (see Fig. 8) would be considered as OTHER errors that were miss-classified, hindering the true performance of the system for the R:VERB category.

3.3.3 Conclusion

ERRANT has definitely provided an alternative, and to some degree, efficient way of annotating datasets for GEC. This is particularly important for GEC systems to be able to assess their own performance and be improved. However, we show that there is still much room for improvement regarding error type classification. Although standardizing corpora can alleviate the annotators from some of the time-consuming labour, incorrect automatic classification might deprive a GEC system from useful information. Especially, in the case of teaching, where automatic feedback is gradually gaining ground, a precise error type classification is mandatory.

In the foreground, more grammar rules should be introduced during the configuration of ERRANT. This will allow a more thorough classification, and therefore more efficient error detection and correction systems. In addition, a qualitative evaluation by linguists could ensure the quality of the classification and provide professional feedback. We release our sample of second order FCE annotations, to pose the ground for the development of a larger reference data set. Potentially, this could be used either as a ground truth evaluation set (e.g., by rule-based systems) or as a training set by more robust machine learning classifiers.

Our next research step would be to delve into the issue of false or ambiguous error type classification further by examining and evaluating more types of errors extracted with ERRANT. We would also like to design a more systematic and thorough error classification system, by employing transfer learning and deep learning approaches.

4 Language Model Prediction

This chapter focuses on the use and effect of language models on grammatical error correction. Although, as mentioned above, GEC methods usually involve transforming erroneous sentences into correct ones, this study attempts a different approach. I focused on whether language models can predict efficiently the correction token in a given sentence, in order to investigate whether they are also mistaken when they are asked to generate tokens that the ESL learners got wrong. For this purpose, I used two different language models: a statistical language model and a deep neural language model. The next chapter is dedicated to putting predictive text into practice, as I experimented with real English language learners and evaluated whether the learners can be assisted by the language model.

4.1 Data Preparation

As mentioned in the Data Analysis section (Table 4), the data was in M2 format. For that reason, and to train the language models, pre-processing was mandatory. For the train data sets, the procedure involved using the corrections to re-create a corrected text. I opted for correcting, and later using for training and evaluation, only the sentences that contained R:PREP errors. This decision had to do with time limitations, since each error type required a different pre-processing approach, as well as with whether the language model will be better trained in prepositions than other words. Sentences with more than one errors were also eliminated for the same reasons. All data sets can be found in the BEA-2019 shared task website.[14] Write and Improve+LOCKNESS, and FCE data sets can be downloaded directly, while Lang-8 and NUCLE can be obtained upon request. The data are provided both in M2 and JSON formats. JSON format is the raw version and does not include any correction edits. After importing the data into the colab notebook, I parsed it and created a Pandas dataframe (Wes McKinney, 2010; pandas development team, 2020).

It has to be noted that in the M2 format, the offsets were in the same column as strings and they had to be seperated as well as converted into integers. This will later help with slicing the tokenized sentence and replacing the wrong preposition with the correct one. After successfuly creating the dataframe, I filtered that data and kept only the sentences that contained errors. Next, the sentences that contained more than one errors were dropped, and they were filtered once more to keep the R:PREP ones. Then, I re-indexed the dataframe. For the cleaning procedure, I kept stopwords because, if removed many of the prepositions are also removed (e.g. of). Then I proceeded to tokenization using the Natural Language Toolkit (NLTK) (Loper & Bird, 2002). For that, punctuation was also kept because it would mess with the token range and therefore the offsets. Finally, I proceeded with the replacement. I lower-cased all words for harmonization, and saved the output in .txt form, as the example in Table 11.

For the test set, I used a similar procedure. I used Pandas to turn the parsed data into a dataframe as in Table 10. From there, I used the offsets to slice the sentence just before

[14] https://www.cl.cam.ac.uk/research/nl/bea2019st/data

Table 10

Pandas dataframe example

Index	sentence	Starting_off	Ending_off	word	tokenized_sents
0	I am writing in order to express my disappointment about your musical show Over the Rainbow	9	10	with	[I, am, writing, in, order, to, express, my, disappointment, about, your, musical, show, Over, the, Rainbow]

Note. The dataframe contains 6 columns. From left to right is the index. This is not the initial index existing in the data before the pre-processing. It is a new index that helps with the location of the sentences in the dataframe. Next is the sentence column which contains all the erroneous sentences used, followed by the offsets. The starting offset indicates the token with which the error starts. The index of the first token of the sentence is zero (0). Ending offset is the token just before the last token of the error span. Second column from the right is the correction word that contains the correct preposition. Finally, last column is the tokenized erroneous sentence.

Table 11

Pre-processed data sample

i am writing in order to express my disappointment with your musical show Over the Rainbow
the problems started at the box office where we asked for the discounts you announced in the advertisement and the man who was selling the tickets said that they did nt exist
on the other hand the theatre restaurant was closed for unknown reasons
her friend Pat had explained the whole story to her husband

Note. Each sentence occupies a line.

the first error token. Finally, I saved the new sliced sentences into a separate dataframe.

4.2 Language Models

The first language model or LM I used in this study was a statistical one (Pavlopoulos & Papapetrou, 2020).[15] Generally, statistical language models are based on the probability of all possible sequences of words, and more specifically by assigning a probability to each possible next word (Jurafsky & Martin, 2009). This particular one uses N-gram statistics, the simplest way to assign probability, to predict and generate the next word. The term N-gram is defined as the sequence of N words. For example bigram (a two-word sequence) can be "your homework", and a trigram (a three-word sequence) can be "turn your homework" (Jurafsky & Martin, 2009). For this SLM, I used bigrams. For the data, after importing all the necessary data sets, into the colab notebook, I merged them together to create an accumulative training set. The training set was also tokenized. After

[15] https://github.com/ipavlopoulos/lm

importing the pre-processed test set in CSV. format, I read it with Pandas and I applied the method that enables the generation of the next gram onto the column with the sliced sentences. The output of the method would be saved into a separate column in the data frame. You can see an example below in Table 12. Finally, I evaluated the language model by calculating its accuracy.

Table 12

Pandas dataframe test data example

Index	sentence	Starting_off	Ending_off	word	prediction
0	I am writing in order to express my disappointment	9	10	with	with

The second language model I used was GPT-2 (Radford et al., 2019). GPT-2 is based on the GPT model of the OpenAI (Radford, 2018). The aim of this model was to be able to perform multiple tasks unsupervised and without requiring a manual training set creation and annotation. The final system was a 1.5B parameter Transformer. During its creation, GPT-2 was tested on multiple tasks. The final verdict underlined that it excels at reading comprehension yet there is still room for improvement in other tasks such as summerization. Of course, I used GPT-2 for language generation. For this purpose, I fine-tuned GPT-2 [16] by training it with part of the BEA-2019 data (see. Chapter 3). I then tested the model in the same way as the SLM.

4.3 Results and Discussion

The first part of the study involved using the the language models to predict the next token of the sliced sentences, and measuring the accuracy against the gold references, which were extracted from the initial M2 file. The second stage involved predicting a greedy selection of 3 predictions and checking if the gold reference is among them, and whether this can elevate the accuracy of the model. As mentioned above, the sentences were sliced just before the erroneous token (see Table 12). The results are presented in the table below:

Table 13

Accuracy scores of SLM and GPT-2.

Model	**Top prediction**	**Top 3 predictions**
SLM	13%	21%
GPT-2	17%	26%

As expected, GPT-2 did better in predicting the correct token, however not by much. Out of 225 sliced sentences of the test data set, both models managed to correctly predict

[16] https://github.com/huggingface/transformers

15% of the correct tokens. Despite its low value, what this percentage tells us is that in 15% of the cases that a word was mistaken, the model would have suggested a correct one for the respective learner that made the mistake. It remains unknown though if the learner would have chosen the correct word or if the system could lead to more errors overall. The accuracy does indeed elevate (20-25%) when the model gives 3 predictions and one of them can be mapped to the correct token (a user, for example, will see 3 options to choose the next word from).

To fully investigate the capacity of predictive text, I conducted a secondary experiment where I sliced the sentences before words that were not mistaken by the learners, and I used the two language models to generate the next token. Both language models did not perform very well in this task, with the estimated accuracy being less than 4%. Elimination experiments that involved variations of the parameters in the language models, or using other corpora for training, did not seem to have a great effect on the performance of the language models.

One possible explanation for these scores lies in the nature of the test set. In the first experiment, the language models have to predict the correct preposition. Prepositions can generally be encountered more frequently than other POS, therefore, the systems see them more often during training and learn their patterns and pairings more efficiently. In addition, prepositions are more possible to appear in a bigram than other words, thus the prediction of preposition bigrams is easier. The task becomes much more difficult when the system has to predict the continuation of words that do not belong to the same POS, and especially when the correct token might be a word that has not been encountered by the system during training. Manual evaluation would give higher scores, because the suggestions of the system are not always mistaken, but they are simply not the ones of the corrections. That is, there may be more than one correct answers per error. This issue is illustrated in the following Table 14.

Table 14

GPT-2 fails to predict the correct token.

Sentence	**Correct Token**	**Prediction 1**	**Prediction 2**	**Prediction 3**
So you are going to come	at	home	across	into

When it comes to preposition prediction, the performance of the language models could be further improved. Elghafari, Meurers, and Wunsch (2010) achieved a 76.5% accuracy with their surface-based n-gram strategy, emphasising, nonetheless, that there are still issues to tackle, such as the nature of the preposition, i.e., whether it is functional or lexical. To correctly predict a functional preposition, one only needs the context (e.g. Mary is dependant **on** her phone), while for a lexical preposition prediction context is not enough (e.g. He put the box **on/under/behind** the table).

To sum up, next word gram generation definitely has potential regarding correct token prediction. An optimization of language models can be achieved by taking into account all possible parameters that concern prepositions, (e.g. nature, possible pairings, frequency). Undoubtedly, the task of the correct token prediction becomes all the more difficult when all POS are concerned and more parameters have to be taken into account.

5 Using predictive text in ESL

The use of predictive text, not only in education but in daily life (i.e. through messaging) as well, is a controversial issue. On the one hand, one could argue that the automatic completion of sentences might lead to restricting and dulling brain activity, consequently affecting the user's language skills, including their grammar (Waldron, Kemp, Plester, & Wood, 2015). On the other hand, there are those who advocate that predictive text systems might in fact enhance the user's ability to generate more creative texts (Waldron et al., 2017). Such a result could therefore mean that predictive text can help acquiring a better command of the language. To explore the potential of predictive text in regards to second language learning, and more specifically to ESL, I conducted an experiment with real English language learners. The goal of the experiment was to determine whether predictive text is beneficial for L2 learners. In this chapter,

- I provide certain information on the language model used for the experiment.
- I provide information about the participants.
- I explain the instructions given before the experiment.
- After the experiment, I mention how the data obtained was processed.
- Finally, I conduct a data analysis along with some qualitative manual observations, and make deductions.

All the above are taken into account in an attempt to answer the question whether the predictive text tool can aid ESL learners in their writing.

5.1 Method

5.1.1 AllenNLP Platform

The most effective language model currently is the OpenAI's unsupervised transformer language model GPT-2. GPT-2 can generate human-like discourse to the degree that it is regarded as a dangerous tool, due to the fact that it can be used for malicious purposes, such as generation of fake news. GPT-2 documentation provides a large variety of parameters and its abilities vary from text summarization to dialogue generation. For the purposes of this study, I opted for a system modification that provides the user/learner with more than one predictions. Thankfully, there is a platform that provides a demonstration of the ability of GPT-2 to predict more than one possible words in the AllenNLP website.[17] The platform uses the public 345M parameter of GPT-2 to generate sentences. In this platform, while the user is writing any sort of text, the top 10 predictions appear next to the writing prompt in the form of sentences.

[17] https://demo.allennlp.org/next-token-lm?text=AllenNLP%20is

Figure 9: AllenNLP platform

5.1.2 Participants

The participants of this study were two English language learners. Both of them have B2[18] certification in the English language. The first participant is a 19-year-old female university student from Greece, currently studying English to get higher certification. She has not stopped taking English classes since the age of 9. The second participant is a 50-year-old female. She had not had English classes for several years until she decided to start again for her own reasons.

5.1.3 Instructions

The experiment was conducted in three phases. During the first phase, the participants were instructed to write three compositions. They were presented with an array of 8 topics taken from the First Certificate in English past papers, and they could choose whichever topic they wanted. They were instructed to write without any additional tools (e.g. translation tools, dictionaries, or asking for help). The word limit was 140-190 words, as in the examination. However, there was a leeway of 50 words. To write the composition they both used Microsoft Word, hence the essays were in document form. For the second phase, participants were instructed to choose from the remaining topics. They would repeat the same procedure. However, this time they wrote the essay on the AllenNLP demo platform where they were allowed to use the sentence suggestions on the right, whenever they saw fit. The third phase involved a small discussion on their experience with the platform. Before the beginning of the experiment both participants filled an ethics form.[19]

5.1.4 Data Preparation

As mentioned above, the 12 essays were initially in document form and were later converted into text form. In the text document, each new sentence occupies a separate line. This is a mandatory step because such format is required for the ERRANT (see chapter 3.3) tool to work. ERRANT feeds on the original and corrected sentences and produces an M2

[18] https://www.coe.int/en/web/common-european-framework-reference-languages/level-descriptions
[19] It can be found in the Appendix

output (Table 4). The next step was to put the data into ERRANT. From the output of ERRANT, the error types were saved into 4 lists. Two lists for each learner, one with the error types made when using the tool, and one without it. This will enable the calculation of the total of the error types and the comparison between the essays written with the predictive text tool and those written without it.

5.2 Results and Discussion

The results of the experiment confirmed the initial hypothesis that predictive text can help the learner to some degree, but the success is also quite dependent on the learner's individual characteristics. Noteworthy is the fact that both participants chose almost exactly in the same way which topic to write with the tool and which without it. Some initial statistics show both participants wrote around 1000 words in total, yet Learner A made almost twice as many errors as learner B. In addition, the word count of the essays of Learner A seem to fluctuate much more, with the longest essay being 215 words and the shortest 117 words.

Table 15

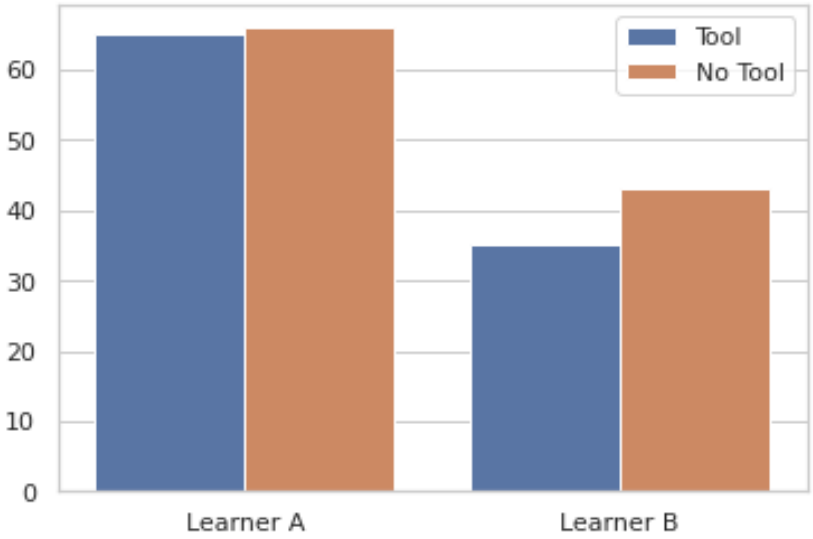
Basic statistics

	Learner A		Learner B	
Topic	Total Words	Total Errors	Total Words	Total Errors
1	150	10.6%	152	6.5%
2	215	16.7%	150	12%
3	117	11.9%	143	5.5%
4	215	14.4%	160	8.8%
5	168	9.5%	202	9.9%
6	160	11.9%	186	4.8%
Total	1025	12.9%	993	8%

Note. The rows in gray indicate that those topics were written with the predictive text tool. Apart from the first topic, which was different, both learners chose in the same way.

Looking at Figure 10, it is apparent that predictive text yielded different results for each learner. Learner A did not show any significant improvement with the tool, whereas Learner B has made fewer mistakes. The question that follows is what does the outcome depend on? In an attempt to answer this question, it is worth to look at some of the participants' characteristics. Learner A is a 50 year old female without much contact with technology, while Learner B is 19-year-old student, probably having spent a lot of time on her computer or any other electronic device since a young age. Taking this into account, we must think that older people do not have the same familiarity with technology as

Figure 10: Number of errors with and without using the predictive text tool.



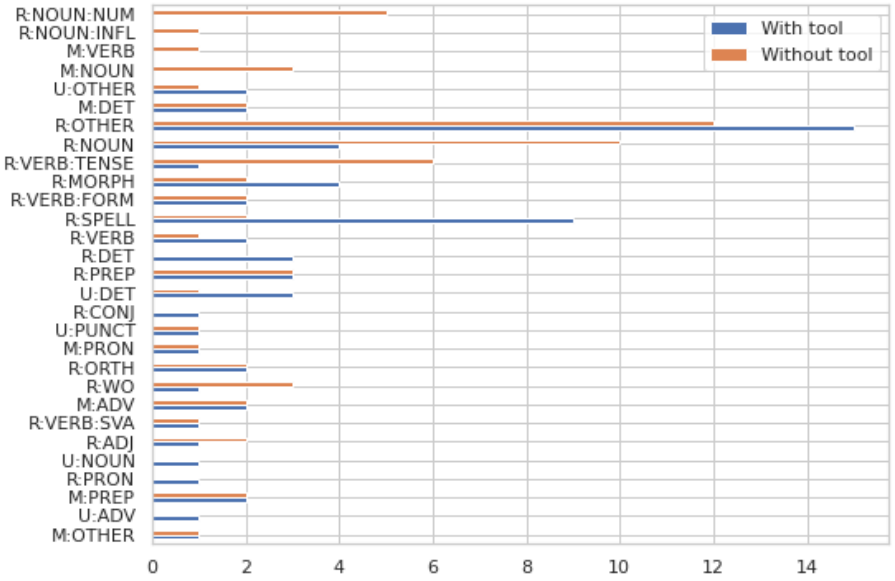
young people. The fact that they have to type the essay instead of writing it and then also clicking on the most appropriate continuation of the sentence might be a challenge for them. On the other hand, younger people are expected to be facilitated from such applications and platforms, given that they can very quickly pick up how to use them. This also confirms the findings in Y. Kalman et al. (2015), in that age and cohort effects can influence once ability to use predictive text.

As far as the error types the learners made are concerned, looking at Figure 11, the first thing that we see is that the most common error type made when using the predictive text tool is R:OTHER, with almost 15 out of the 66 errors. This is not however the true portion of the classification of the errors, because, as examined in chapter 3.3, the automatic annotation tool has a major shortcoming, which prevents a more accurate error type classification. The second, and third most frequent error types are R:SPELL and R:NOUN along with R:MORPH, respectively. The rest of the error types presented less than 4 occurrences in total. Without using the predictive text tool, the most frequent error types is again R:OTHER, however, with twice fewer occurrences than with the tool. R:NOUN comes in second position with 10 occurrences, while R:VERB:TENSE comes in third with 6 occurrences. R:NOUN:NUM errors where also frequent, appearing 5 times in this sub-data set.

The most frequent error type Learner B made without the tool was R:NOUN, with 8 occurrences. R:OTHER was the second most frequent with 7 occurrences. R:OTHER was the most frequent error types with tool, as well. Spelling (R:ORTH), along with determiner U:DET mistakes were quite frequent, too, when the tool was used.

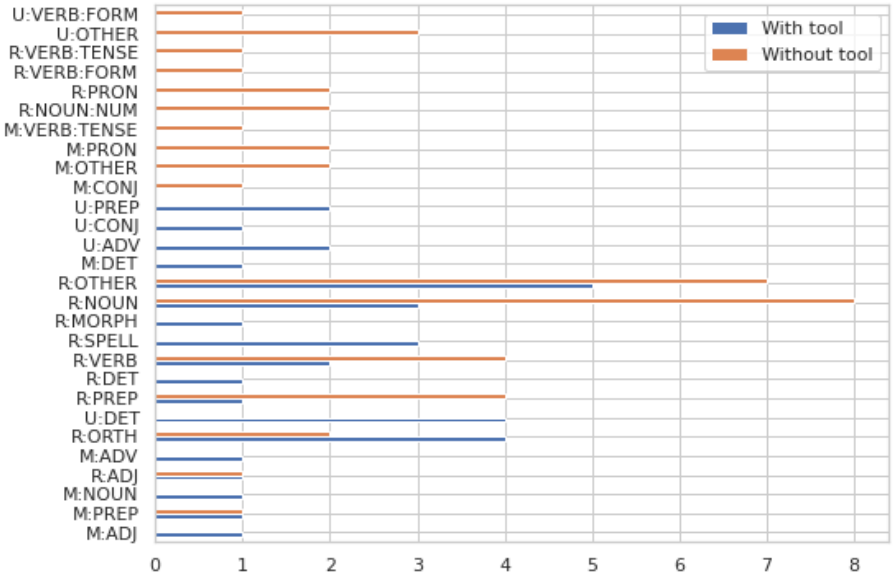
An observation that could be made when comparing the two figures concerns the consistency of error types, when switching from no tool to tool. Particularly, Learner A made mostly the same types of errors with and without the tool, with only 9 not overlapping. On the other hand, the error type pattern of learner B is completely different with only 7 error types overlapping, meaning that, although fewer mistakes have been made, there

Figure 11: Learner A error types before and after the use of the predictive text tool.



Note. Learner A made 24 different errors types with the tool and 24 without the tool.

Figure 12: Learner B error types before and after the use of the predictive text tool.



Note. Learner B made 17 error types without the tool and 18 error types with the tool.

is a greater variation of error types and less consistency. A deduction that could be made from this observation is that Learner A wrote the essays as she would without using the tool, and relying on the suggestions as little as possible. However, this could also mean that some errors of learner A can be hiding in the OTHER category.

After the completion of the task, the two learners shared their thoughts about their experience of the predictive text tool. Learner B was very supportive of the use of such tools in class. She claimed that the tool helped her write much faster and that she wished that she could use it during examinations. She also underlined that even though the tool presented some "ready-to-use" sentences, she could learn from it because it suggested syntactical combinations and vocabulary that she had not encountered before. She also commented very positively on the time-saving benefit of the tool. Learner B, on the other hand said that although she did not find the tool confusing to use, she found the process of using it time consuming. This is a very interesting comment given that Learner B used to know how to write in blind system, but still considered typing the essay time-consuming. She also complained that it sometimes "lagged".

In conclusion, this study demonstrated a way to use a predictive text tool for second language learning purposes. The results did not deviate from previous studies that examined the use of predictive text among different age groups. On the contrary it confirmed them. An extended version of the experiment with more participants and time profiling would provide a more integrated picture of how predictive text can aid second language learning.

6 Conclusion

This thesis has examined the potential of language modelling and, in particular, predictive text generation, in Grammatical Error Correction for Second Language Learning. The study was developed in two stages. The first stage involved evaluating the accuracy of a statistical language model and an unsupervised neural language model regarding the prediction of the correct token in erroneous sentences. The evaluation of those language models showed that they can reach from 15% to 25% accuracy suggesting the correct token in a sentence that contains a mistake. The second stage was an experiment with real ESL learners where they were called to write essays with and without using a predictive text tool. The experiment confirmed the findings of previous studies that elicited that predictive text tools, as well as autocorrect can help in the reduction of grammatical mistakes in the learners writing. However, this is also dependent on the learners personal characteristics and cohort effects, such as age and familiarity with technology.

This thesis has opened several fronts in GEC. A first future endeavour would be to improve the ERRANT classification process by using a deep learning approach. This will also contribute to the development of more efficient GEC systems. Moreover, creating a predicting text tool, specific to L2 learning, and testing it in a classroom setting and by also using a control group, will provide a more detailed image on whether predictive text can assist ESL.

References

Ailani, S., Dalvi, A., & Siddavatam, I. (2019, 08). Grammatical error correction (gec): Research approaches till now. *International Journal of Computer Applications*, *178*, 1-3. doi: 10.5120/ijca2019919275

Arnold, K. C., Chauncey, K., & Gajos, K. Z. (2020). Predictive text encourages predictable writing. In *Proceedings of the 25th international conference on intelligent user interfaces* (p. 128–138). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3377325.3377523 doi: 10.1145/3377325.3377523

Boyd, A. (2018, November). Using Wikipedia edits in low resource grammatical error correction. In *Proceedings of the 2018 EMNLP workshop w-NUT: The 4th workshop on noisy user-generated text* (pp. 79–84). Brussels, Belgium: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/W18-6111 doi: 10.18653/v1/W18-6111

Bryant, C., & Briscoe, T. (2018, June). Language model based grammatical error correction without annotated training data. In *Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications* (pp. 247–253). New Orleans, Louisiana: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/W18-0529 doi: 10.18653/v1/W18-0529

Bryant, C., Felice, M., Andersen, Ø. E., & Briscoe, T. (2019, August). The BEA-2019 shared task on grammatical error correction. In *Proceedings of the fourteenth workshop on innovative use of nlp for building educational applications* (pp. 52–75). Florence, Italy: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/W19-4406 doi: 10.18653/v1/W19-4406

Bryant, C., Felice, M., & Briscoe, T. (2017, July). Automatic annotation and evaluation of error types for grammatical error correction. In *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 793–805). Vancouver, Canada: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/P17-1074 doi: 10.18653/v1/P17-1074

Dahlmeier, D., & Ng, H. T. (2012, June). Better evaluation for grammatical error correction. In *Proceedings of the 2012 conference of the north American chapter of the association for computational linguistics: Human language technologies* (pp. 568–572). Montréal, Canada: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/N12-1067

Dahlmeier, D., Ng, H. T., & Wu, S. M. (2013, June). Building a large annotated corpus of learner English: The NUS corpus of learner English. In *Proceedings of the eighth workshop on innovative use of NLP for building educational applications* (pp. 22–31). Atlanta, Georgia: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/W13-1703

Davidson, S., Yamada, A., Fernandez Mira, P., Carando, A., Sanchez Gutierrez, C. H., & Sagae, K. (2020, May). Developing NLP tools with a new corpus of learner Spanish. In *Proceedings of the 12th language resources and evaluation conference* (pp. 7238–7243). Marseille, France: European Language Resources Association. Retrieved

from https://www.aclweb.org/anthology/2020.lrec-1.894

Elghafari, A., Meurers, D., & Wunsch, H. (2010, August). Exploring the data-driven prediction of prepositions in English. In *Coling 2010: Posters* (pp. 267–275). Beijing, China: Coling 2010 Organizing Committee. Retrieved from https://www.aclweb.org/anthology/C10-2031

Felice, M., Yuan, Z., Andersen, Ø. E., Yannakoudakis, H., & Kochmar, E. (2014, June). Grammatical error correction using hybrid systems and type filtering. In *Proceedings of the eighteenth conference on computational natural language learning: Shared task* (pp. 15–24). Baltimore, Maryland: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/W14-1702 doi: 10.3115/v1/W14-1702

Gajos, K. Z., Hurst, A., & Findlater, L. (2012, March). Personalized dynamic accessibility. *Interactions*, *19*(2), 69–73. Retrieved from https://doi.org/10.1145/2090150.2090167 doi: 10.1145/2090150.2090167

Granger, S. (1998). The computerized learner corpus: a versatile new source of data for sla research..

Granger, S. (2003, 01). Error-tagged learner corpora and call: A promising synergy. *CALICO Journal*, *20*, 465-480.

Grundkiewicz, R., & Junczys-Dowmunt, M. (2018, June). Near human-level performance in grammatical error correction with hybrid machine translation. In *Proceedings of the 2018 conference of the north American chapter of the association for computational linguistics: Human language technologies, volume 2 (short papers)* (pp. 284–290). New Orleans, Louisiana: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/N18-2046 doi: 10.18653/v1/N18-2046

Jurafsky, D., & Martin, J. H. (2009). *Speech and Language Processing : an introduction to natural language processing, computational linguistics, and speech recognition*. Pearson Prentice Hall/Pearson education international. Retrieved from http://books.google.de/books?id=crxYPgAACAAJ

Kaan, E. (2014). Predictive sentence processing in l2 and l1: What is different? *Linguistic Approaches to Bilingualism*, *4*, 257-282.

Kalman, Y., Kavé, G., & Umanski, D. (2015, 10). Writing in a digital world: Self-correction while typing in younger and older adults. *International journal of environmental research and public health*, *12*, 12723-12734. doi: 10.3390/ijerph121012723

Kalman, Y. M., Geraghty, K., Thompson, C. K., & Gergle, D. (2012). Detecting linguistic hci markers in an online aphasia support group. In *Proceedings of the 14th international acm sigaccess conference on computers and accessibility* (pp. 65–70).

Kaneko, M., Mita, M., Kiyono, S., Suzuki, J., & Inui, K. (2020). *Encoder-decoder models can benefit from pre-trained masked language models in grammatical error correction.*

Kantor, Y., Katz, Y., Choshen, L., Cohen-Karlik, E., Liberman, N., Toledo, A., ... Slonim, N. (2019). Learning to combine grammatical error corrections. *CoRR*, *abs/1906.03897*. Retrieved from http://arxiv.org/abs/1906.03897

Katsumata, S., & Komachi, M. (2019). *Towards unsupervised grammatical error correction using statistical machine translation with synthetic comparable corpus.*

Kiyono, S., Suzuki, J., Mita, M., Mizumoto, T., & Inui, K. (2019, November). An empirical study of incorporating pseudo data into grammatical error correction. In

Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp) (pp. 1236–1242). Hong Kong, China: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/D19-1119 doi: 10.18653/v1/D19-1119

Kraichoke, C. (2017). Error analysis: A case study on non-native english speaking college applicants' electronic mail communications..

Lee, J., & Seneff, S. (2008). An analysis of grammatical errors in non-native speech in english. In *2008 ieee spoken language technology workshop* (p. 89-92).

Loper, E., & Bird, S. (2002). Nltk: The natural language toolkit. In *In proceedings of the acl workshop on effective tools and methodologies for teaching natural language processing and computational linguistics. philadelphia: Association for computational linguistics.*

Meurers, D. (2012). Natural language processing and language learning. In *The encyclopedia of applied linguistics.* American Cancer Society. Retrieved from https://onlinelibrary.wiley.com/doi/abs/10.1002/9781405198431.wbeal0858 doi: 10.1002/9781405198431.wbeal0858

Mutton, A., Dras, M., Wan, S., & Dale, R. (2007, June). GLEU: Automatic evaluation of sentence-level fluency. In *Proceedings of the 45th annual meeting of the association of computational linguistics* (pp. 344–351). Prague, Czech Republic: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/P07-1044

Newell, A., Booth, L., & Beattie, W. (2006, 10). Predictive text entry with pal and children with learning difficulties. *British Journal of Educational Technology*, *22*, 23 - 40. doi: 10.1111/j.1467-8535.1991.tb00049.x

Ng, H. T., Wu, S. M., Briscoe, T., Hadiwinoto, C., Susanto, R. H., & Bryant, C. (2014, June). The CoNLL-2014 shared task on grammatical error correction. In *Proceedings of the eighteenth conference on computational natural language learning: Shared task* (pp. 1–14). Baltimore, Maryland: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/W14-1701 doi: 10.3115/v1/W14-1701

Náplava, J., & Straka, M. (2019). *Grammatical error correction in low-resource scenarios.*

Omelianchuk, K., Atrasevych, V., Chernodub, A., & Skurzhanskyi, O. (2020). *Gector – grammatical error correction: Tag, not rewrite.*

pandas development team, T. (2020, February). *pandas-dev/pandas: Pandas.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.3509134 doi: 10.5281/zenodo.3509134

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics* (p. 311–318). USA: Association for Computational Linguistics. Retrieved from https://doi.org/10.3115/1073083.1073135 doi: 10.3115/1073083.1073135

Pavlopoulos, J., & Papapetrou, P. (2020). *Clinical predictive keyboard using statistical and neural language modeling.*

Radford, A. (2018). Improving language understanding by generative pre-training..

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners.

Richards, J., & Richards, J. (1974). *Error analysis: Perspectives on second language acquisition*. Longman. Retrieved from https://books.google.gr/books?id=lmINAQAAMAAJ

Selinker, L. (1972). Interlanguage. *International Review of Applied Linguistics in Language Teaching*, *10*(1–4), 209–232.

Tajiri, T., Komachi, M., & Matsumoto, Y. (2012, July). Tense and aspect error correction for ESL learners using global context. In *Proceedings of the 50th annual meeting of the association for computational linguistics (volume 2: Short papers)* (pp. 198–202). Jeju Island, Korea: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/P12-2039

Waldron, S., Kemp, N., Plester, B., & Wood, C. (2015, 03). Texting behavior and language skills in children and adults.. doi: 10.1002/9781118771952.ch13

Waldron, S., Wood, C., & Kemp, N. (2017). Use of predictive text in text messaging over the course of a year and its relationship with spelling, orthographic processing and grammar. *Journal of Research in Reading*, *40*(4), 384-402. Retrieved from https://onlinelibrary.wiley.com/doi/abs/10.1111/1467-9817.12073 doi: 10.1111/1467-9817.12073

Wes McKinney. (2010). Data Structures for Statistical Computing in Python. In Stéfan van der Walt & Jarrod Millman (Eds.), *Proceedings of the 9th Python in Science Conference* (p. 56 - 61). doi: 10.25080/Majora-92bf1922-00a

Wu, H., & Garza, E. V. (2014). Types and attributes of english writing errors in the efl context—a study of error analysis. *Journal of Language Teaching and Research*, *5*, 1256-1262.

Yannakoudakis, H., Briscoe, T., & Medlock, B. (2011, June). A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies* (pp. 180–189). Portland, Oregon, USA: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/P11-1019

Yannakoudakis, H., Øistein E Andersen, Geranpayeh, A., Briscoe, T., & Nicholls, D. (2018). Developing an automated writing placement system for esl learners. *Applied Measurement in Education*, *31*(3), 251-267. Retrieved from https://doi.org/10.1080/08957347.2018.1464447 doi: 10.1080/08957347.2018.1464447

Appendix

Table 16

BEA-2019 unrestricted track results

Unrestricted Group	Rank	Teams	TP	FP	FN	P	R	$F_{0.5}$
1	1	LAIX	2618	960	2671	**73.17**	49.50	**66.78**
	2	AIP-Tohoku	2589	1078	2484	70.60	51.03	65.57
2	3	UFAL	2812	1313	2469	68.17	53.25	64.55
3	4	BLCU	**3051**	2007	**2357**	60.32	**56.42**	59.50
4	5	Aparecium	1585	1077	2787	59.54	36.25	52.76
5	6	Buffalo	699	**374**	3265	65.14	17.63	42.33
6	7	Ramaiah	1161	8062	3480	12.59	25.02	13.98

Note. Table from Bryant et al. (2019).

Table 17

BEA-2019 unrestricted track results

Low Resource Group	Rank	Teams	TP	FP	FN	P	R	$F_{0.5}$
1	1	UEDIN-MS	2312	982	**2506**	**70.19**	**47.99**	**64.24**
2	2	Kakao&Brain	**2412**	1413	2797	63.06	46.30	58.80
3	3	LAIX	1443	**884**	3175	62.01	31.25	51.81
	4	CAMB-CUED	1814	1450	2956	55.58	38.03	50.88
4	5	UFAL	1245	1222	2993	50.47	29.38	44.13
5	6	Siteimprove	1299	1619	3199	44.52	28.88	40.17
	7	WebSpellChecker	2363	3719	3031	38.85	43.81	39.75
6	8	TMU	1638	4314	3486	27.52	31.97	28.31
7	9	Buffalo	446	1243	3556	26.41	11.14	20.73

Note. Table from Bryant et al. (2019).

Table 18

All error type categories with explanation and examples

Code	Meaning	Description / Example
ADJ	Adjective	*big → wide*
ADJ:FORM	Adjective Form	Comparative or superlative adjective errors. *goodest → best, bigger → biggest, more easy → easier*
ADV	Adverb	*speedily → quickly*
CONJ	Conjunction	*and → but*
CONTR	Contraction	*n't → not*
DET	Determiner	*the → a*
MORPH	Morphology	Tokens have the same lemma but nothing else in common. *quick (adj) → quickly (adv)*
NOUN	Noun	*person → people*
NOUN:INFL	Noun Inflection	Count-mass noun errors. *informations → information*
NOUN:NUM	Noun Number	*cat → cats*
NOUN:POSS	Noun Possessive	*friends → friend's*
ORTH	Orthography	Case and/or whitespace errors. *Bestfriend → best friend*
OTHER	Other	Errors that do not fall into any other category (e.g. paraphrasing). *at his best → well, job → professional*
PART	Particle	*(look) in → (look) at*
PREP	Preposition	*of → at*
PRON	Pronoun	*ours → ourselves*
PUNCT	Punctuation	*! → .*
SPELL	Spelling	*genectic → genetic, color → colour*
UNK	Unknown	The annotator detected an error but was unable to correct it.
VERB	Verb	*ambulate → walk*
VERB:FORM	Verb Form	Infinitives (with or without "to"), gerunds (-ing) and participles. *to eat → eating, dancing → danced*
VERB:INFL	Verb Inflection	Misapplication of tense morphology. *getted → got, fliped → flipped*
VERB:SVA	Subject-Verb Agreement	*(He) have → (He) has*
VERB:TENSE	Verb Tense	Includes inflectional and periphrastic tense, modal verbs and passivization. *eats → ate, eats → has eaten, eats → can eat, eats → was eaten*
WO	Word Order	*only can › can only*

Note. Table from Bryant et al. (2017).

Figure 13: All error type frequencies in the FCE train corpus.

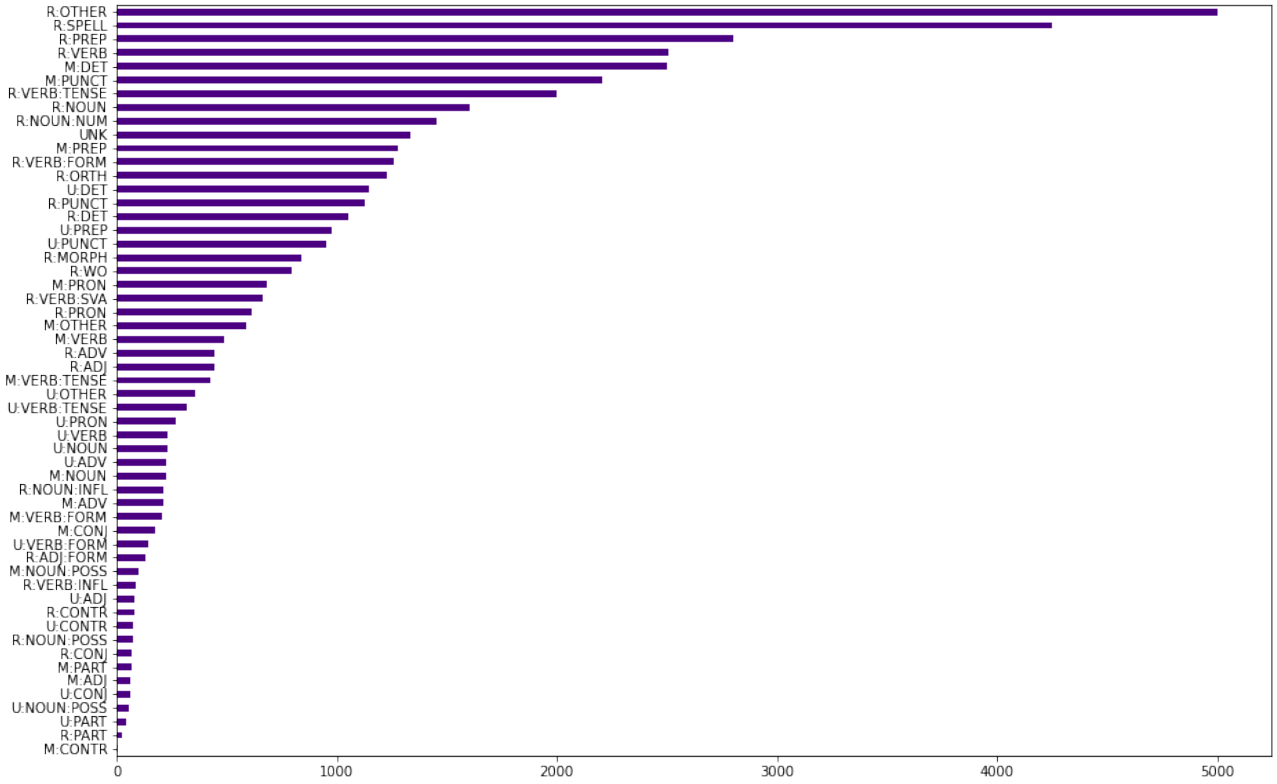


Figure 14: All error type frequencies in the NUCLE corpus.

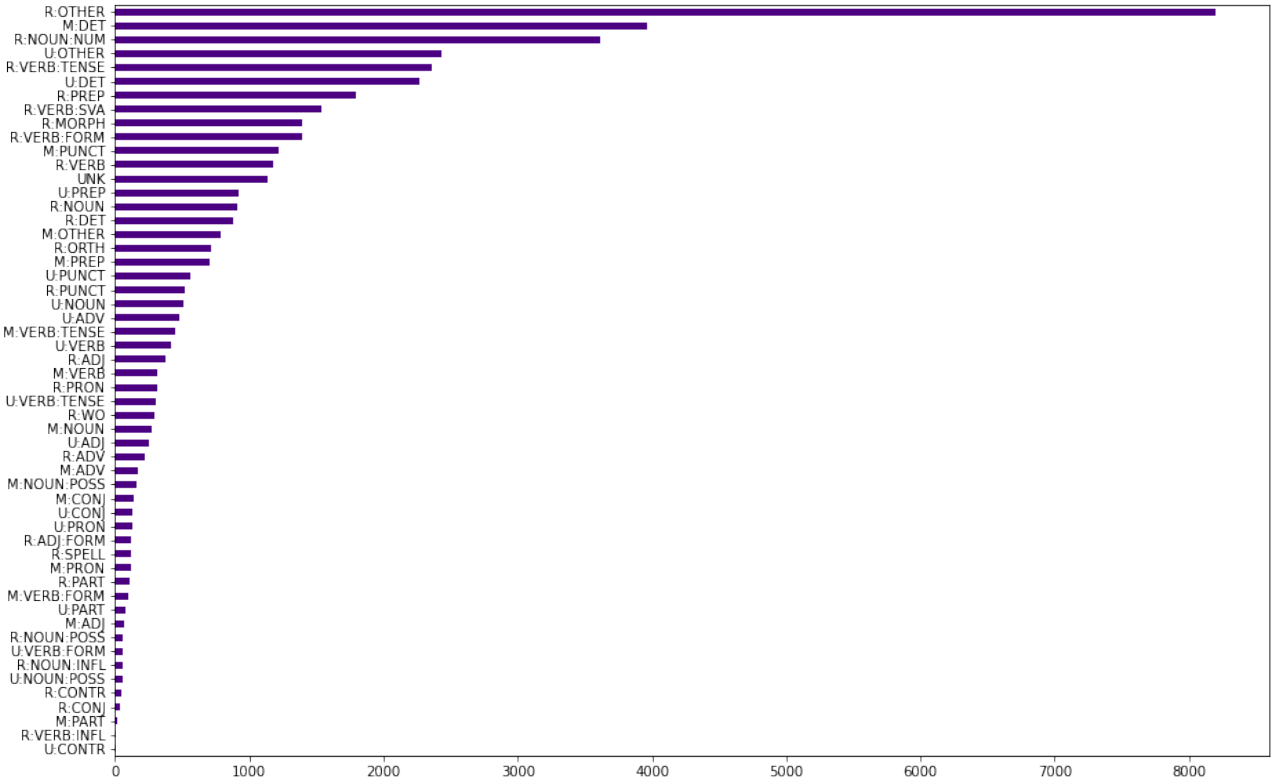


Figure 15: All error type frequencies in the Lang-8 corpus.

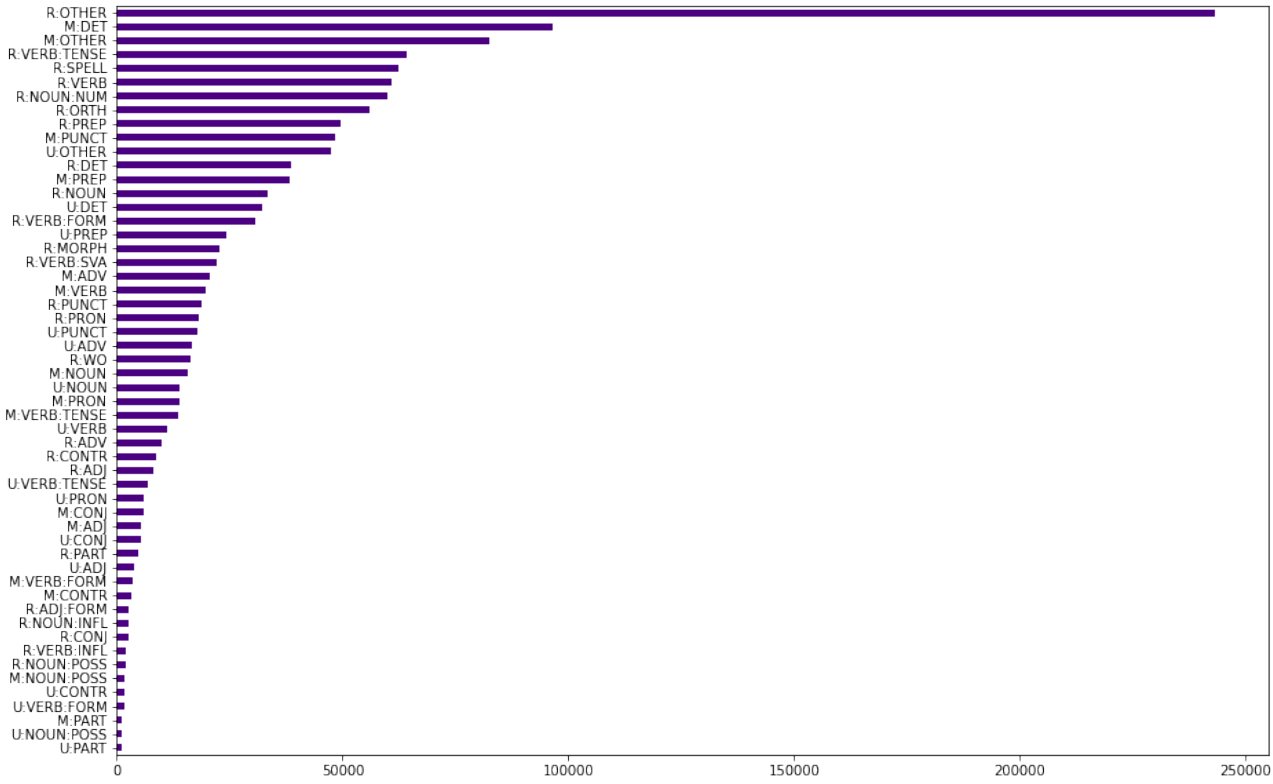


Figure 16: All error type frequencies in the Write & Improve A corpus.

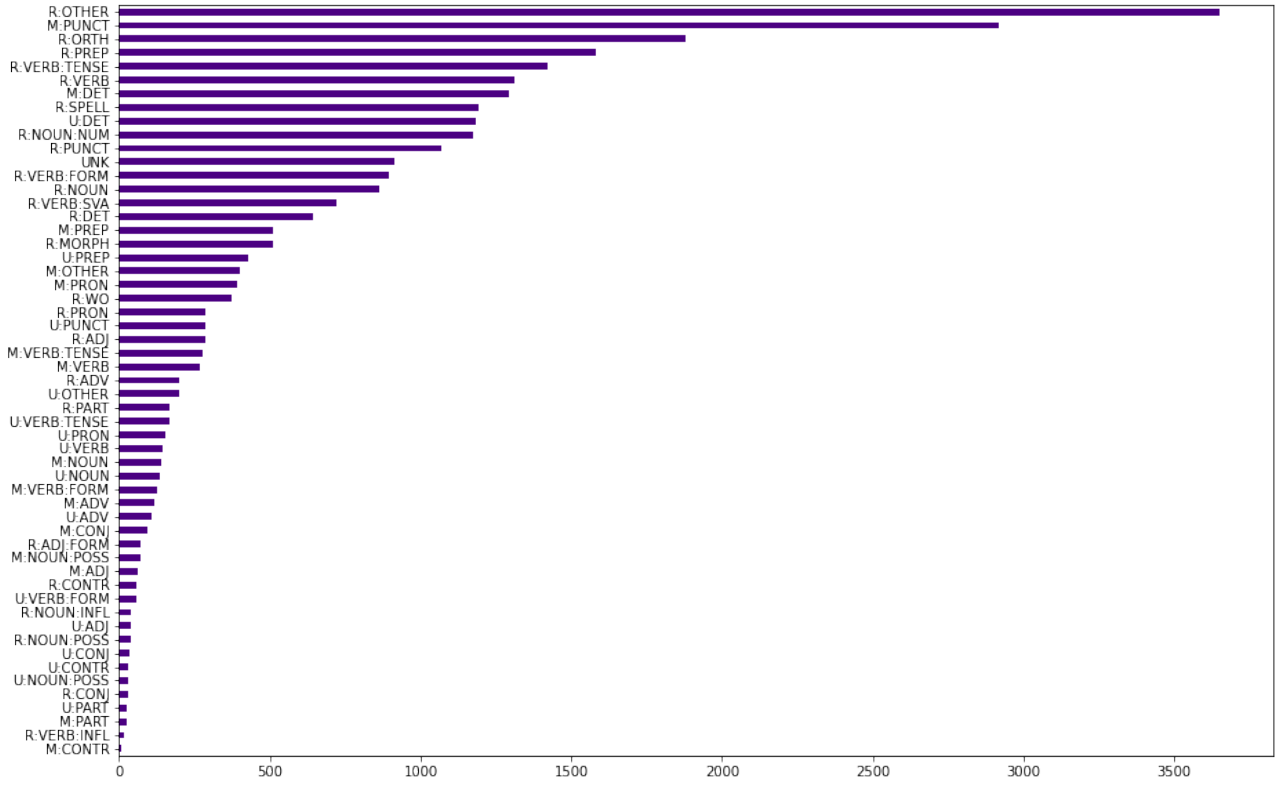


Figure 17: All error type frequencies in the Write & Improve B corpus.

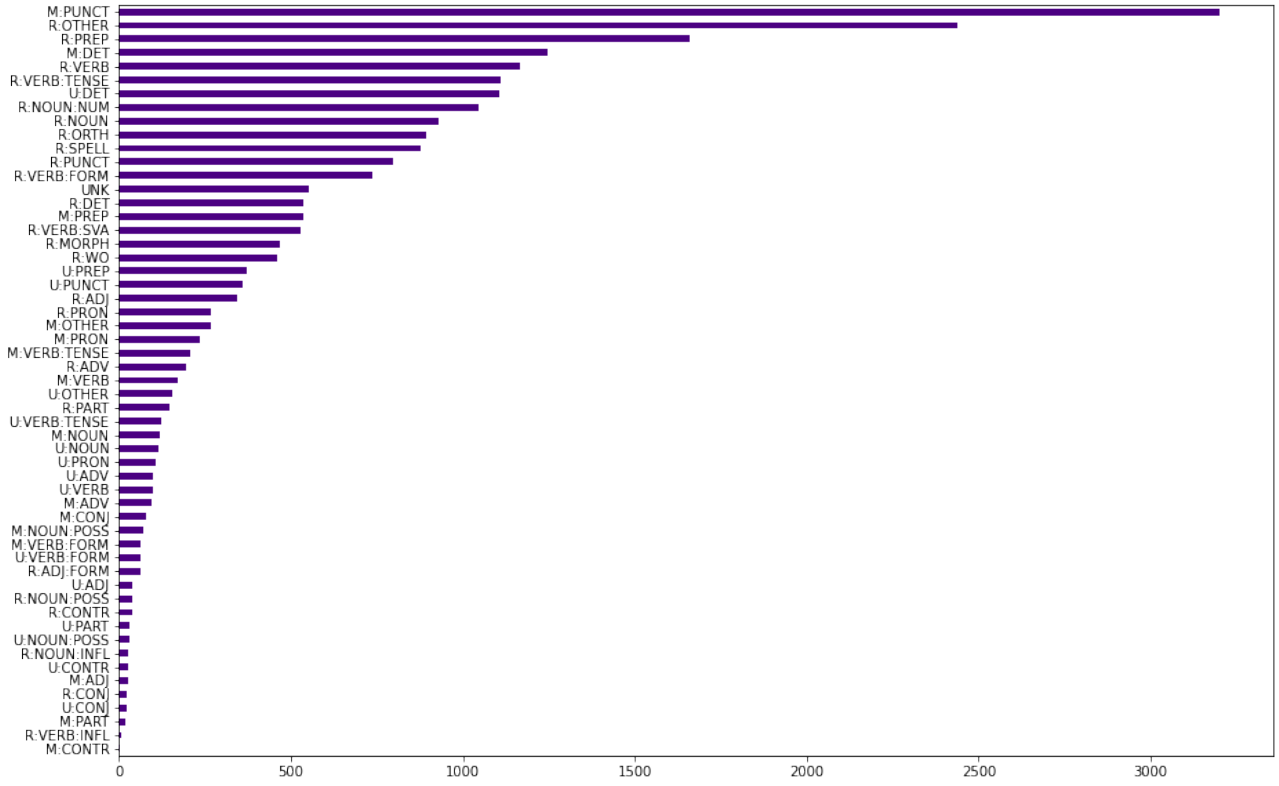


Figure 18: All error type frequencies in the Write & Improve C corpus.

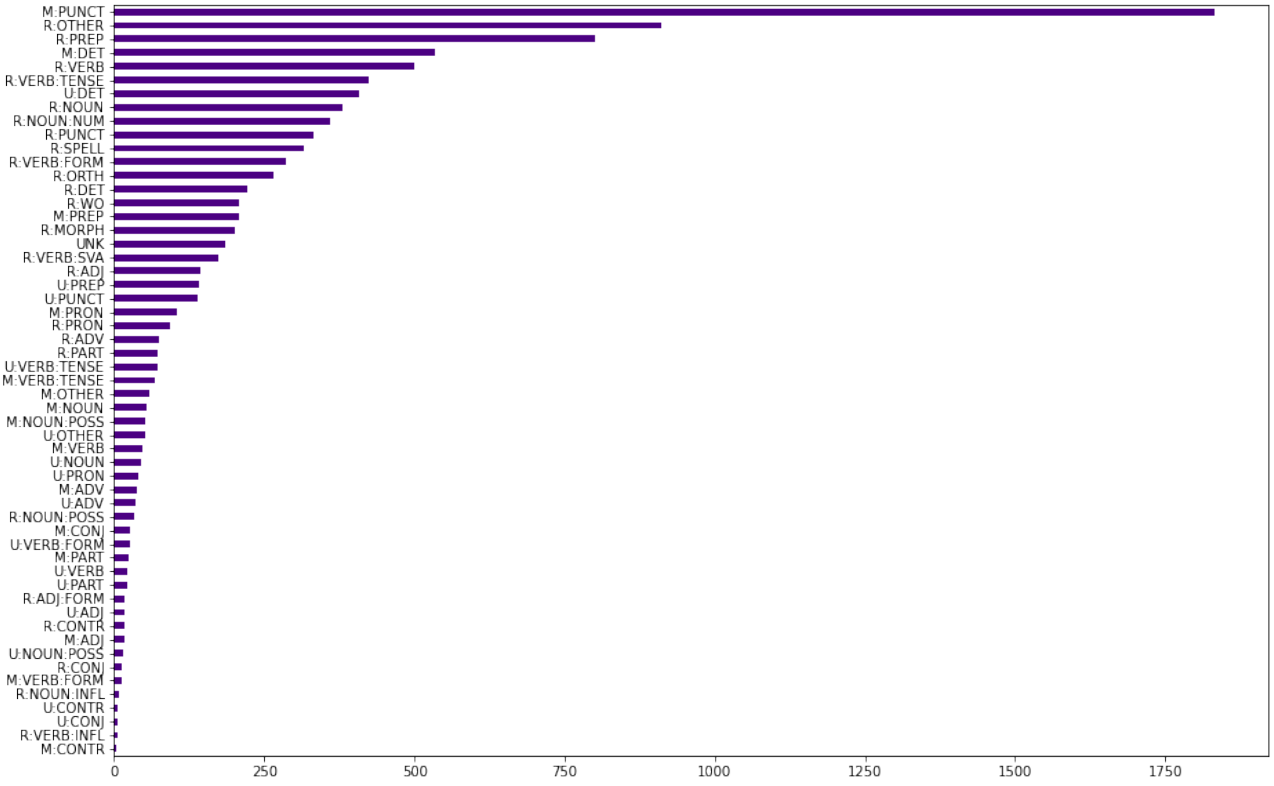


Figure 19: Learner A error type frequencies with tool.

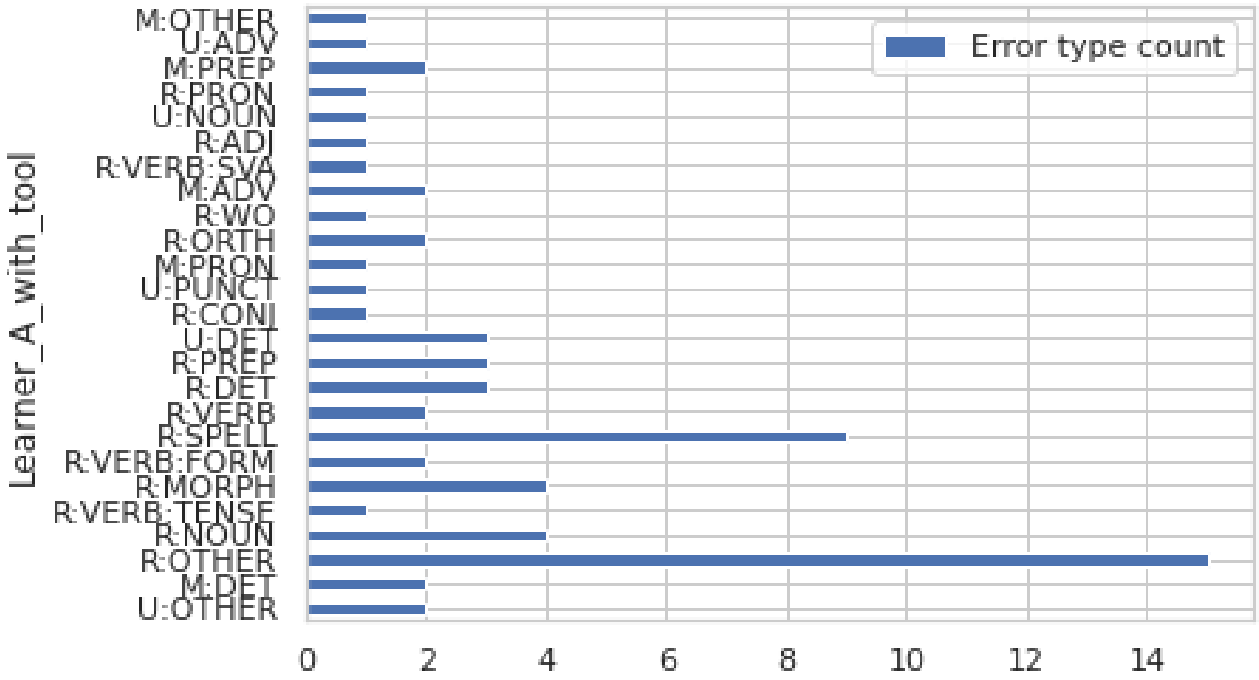


Figure 20: Learner B error type frequencies with tool.

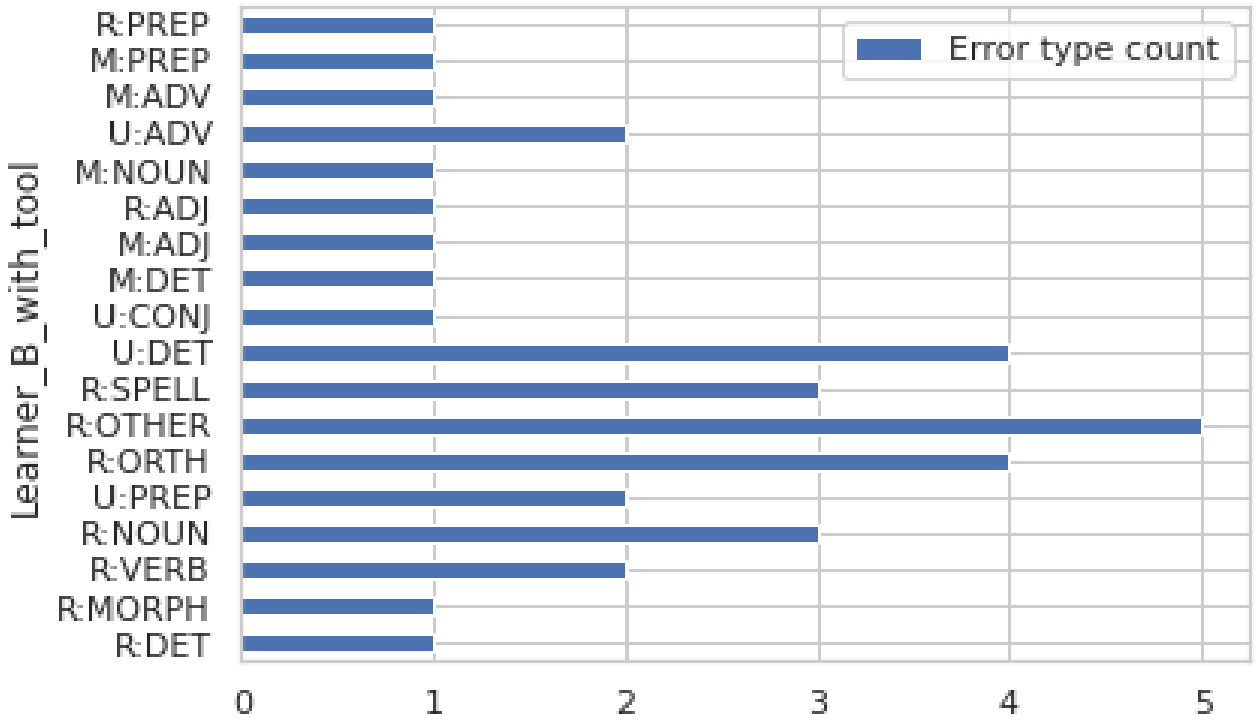


Figure 21: Learner A error type frequencies without tool

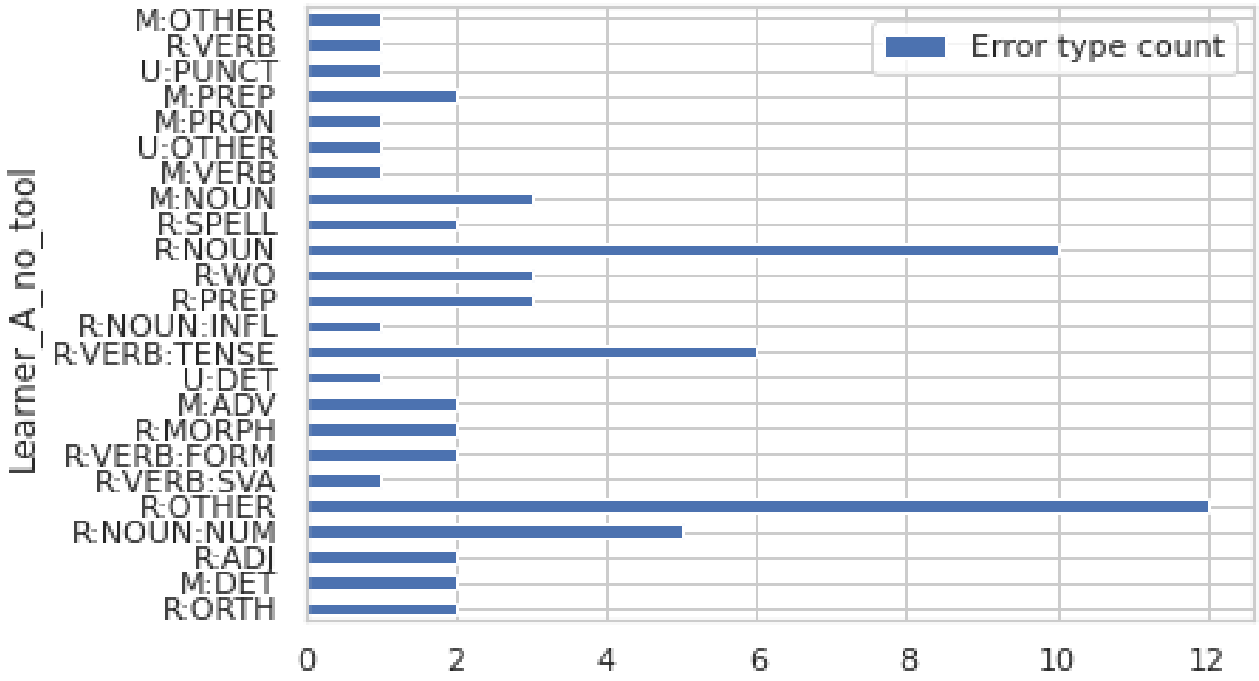
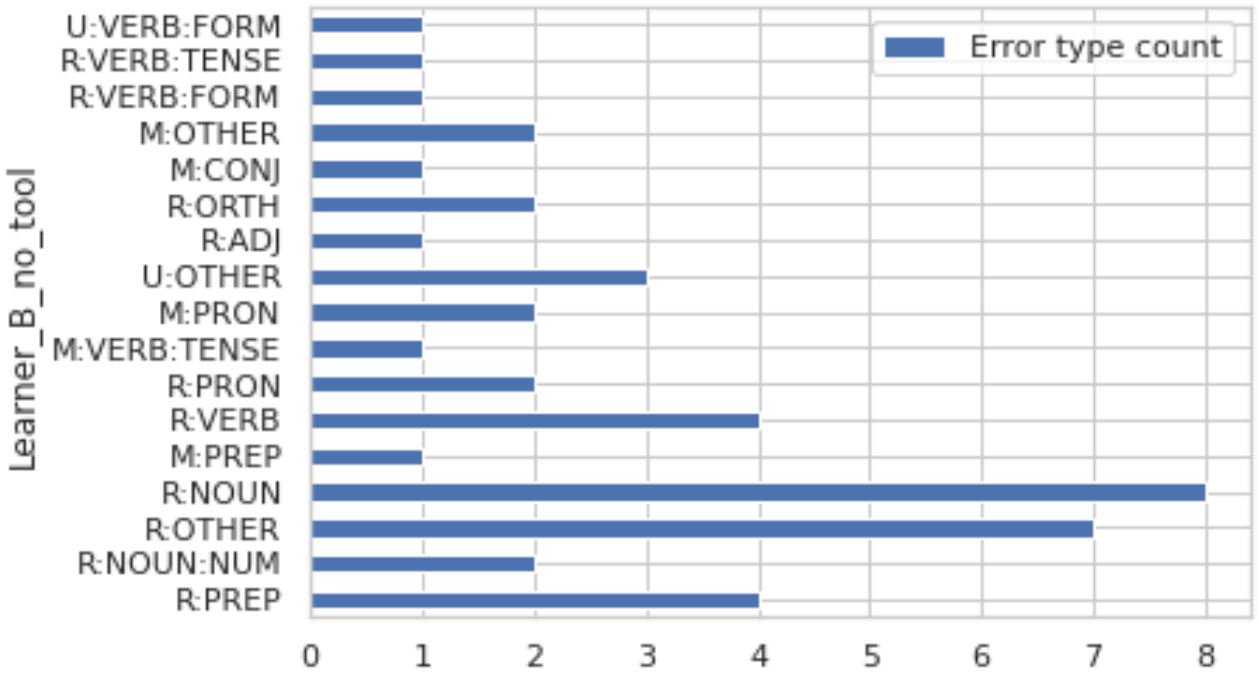


Figure 22: Learner B error type frequencies without tool



PARTICIPANT CONSENT FORM

TITLE OF RESEARCH STUDY: USING PREDICTIVE TEXT IN ESL

PLEASE ANSWER THE FOLLOWING QUESTIONS:

1.	I have fully understood the instructions for this experiment	YES/NO
2.	I understand that I can ask any clarifying question about the process of the experiment, at any time.	YES/NO
3.	I understand that I opt out from the study at any time, without giving any reason for my withdrawal.	YES/NO
4.	I agree to provide information to the researchers provided it stay **confidential**.	YES/NO
5.	I consent that the data collected from my participation be used for any research purposes, once **anonymized.**	YES/NO
6.	I agree to participate in this experiment.	YES/NO

NAME: _____

CONTACT DETAILS: _____

_____

SIGNATURE: _____ **DATE:** _____

RESEARCHER'S NAME: Katerina Korre

RESEARCHER'S SIGNATURE :

RESEARCHER'S CONTACT DETAILS : katkorre95@gmail.com, katkorre95@aueb.gr

Using predictive text in ESL

Experiment Information Sheet

1. **Purpose**
   This is an experiment to determine whether English as a second language (ESL) learners can benefit from using a predictive text tool.

2. **Instructions**
   - Learners fill in a consent form, as well as GDPR form.
   - Learners are presented with a bank of 8 essay topics.
   - Learners choose **three** topics to write **without** the predictive text tool.
   - Learners choose **three** topics to write **with** the predictive text tool.
   - No help is allowed in either of the experiment stages.
   - Small discussion about their experience with the tool.

3. **Notes**
   - LA said she didn't find the tool confusing but found the process more time-consuming than just writing the essays on her own.
   - She complained that it sometimes lagged.
   - She used it more than two times in each essay.
   - Used to know how blind system yet she preferred not using the tool.

- ❖ The tool was very well received by LB. She wrote much faster.
- ❖ She said she wished that she could use this during examinations.
- ❖ She said that she did not know some presented syntactical combinations were feasible and that she felt like she could learn while saving time.