

Mind the Phrase: Annotation and Identification of Multiword Expressions in Grammatical Error Correction

Shiva Taslimipoor¹, Christopher Bryant^{4,1}, Zheng Yuan^{3,1}, Diane Nicholls²,
Andrew Caines¹, Paula Buttery¹,

¹ALTA Institute & Computer Laboratory, University of Cambridge, United Kingdom

²Cambridge University Press & Assessment, United Kingdom

³University of Sheffield, Sheffield, UK

⁴Writer, Inc.

Correspondence: shiva.taslimipoor@cl.cam.ac.uk

Abstract

Multiword Expressions (MWEs) are an important part of language competence which are both commonplace across languages and challenging for language learners to master. Since Grammatical Error Correction (GEC) systems tend to provide suggestions at the token level, they may fail to handle the semantic unity of MWEs. We present annotations of MWEs in learner English essays while identifying their intended spans. This not only allows us to analyse learner MWE errors in more detail for the first time, but also enables us to build GEC systems that provide more meaningful feedback to learners. We have annotated 1,850 essays that were previously tagged for grammatical errors with various types of MWEs. We extend an error annotation toolkit with an MWE error-type and present a method that leverages these annotations to make GEC evaluation more explainable, enabling fine-grained analysis of MWE errors. Experiments with current GEC systems show that an edit-based model fine-tuned on our dataset achieves higher $F_{0.5}$ scores for the correction of common MWE error types, demonstrating its effectiveness for handling MWEs.

1 Introduction

Multiword expressions (MWEs) – combinations of words that function as single semantic or syntactic units – pose significant challenges for language learners due to their unpredictability and often non-compositional meanings. Previous work has highlighted the central role of MWEs in language learning, while underscoring the difficulties they present (Meunier and Granger, 2008; Siyanova-Chanturia and Martinez, 2014; Arnon and Christiansen, 2017). Arnon and Christiansen (2017) argue that language is learned through the acquisition of frequent multiword sequences, which serve as cognitive building blocks for fluent processing and production. Meunier and Granger (2008), drawing on learner corpus data, show that second language learners fre-

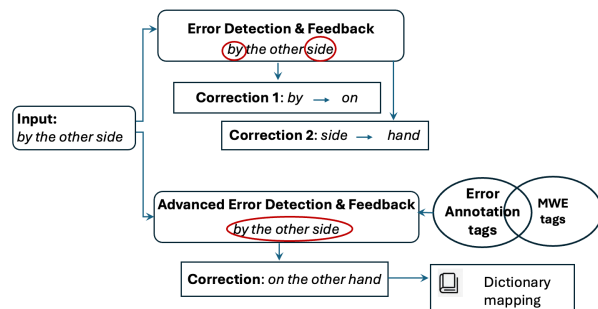


Figure 1: Improving error detection and feedback quality through MWE-informed correction.

quently misuse MWEs, and that these expressions are a persistent source of error and non-proficiency. They argue that formulaic language is essential for achieving fluency and idiomaticity in a second language. Previous work also pointed to the need for pedagogical and computational approaches that better support the acquisition of MWEs (Boers and Lindstromberg, 2008; Paquot and Granger, 2012). Sentsova et al. (2025) re-iterate that high-level language understanding requires the ability to build non-compositional expressions.

Grammatical Error Correction (GEC) is the task of automatically detecting and correcting errors in texts, where a system takes an erroneous sentence as input and produces a grammatically correct output. Language learners can be creative in generating idiomatically incorrect phrases such as *by the other side* or *in the other hand* rather than *on the other hand*. GEC systems, however, typically treat these cases as minimal token-based edits, such as *by* → *on*, *side* → *hand*, and *in* → *on* and so fail to capture the full extent of the error. This behaviour is further compounded by the fact that the most commonly used GEC evaluation tools perform at the token level (e.g., M² scorer Dahlmeier and Ng, 2012, ERRANT, Bryant et al., 2017). We thus propose that GEC systems should go beyond the token level, when handling MWEs (see Figure 1).

Mizumoto et al. (2015) and Taslimipoor et al.

(2022) show that if GEC systems are fed with MWE tags, their performance improves for some categories of MWEs. The MWEs in these works are automatically detected.

Recent GEC systems fall into three main categories: sequence tagging, sequence-to-sequence (seq2seq), and LLM-based approaches. Sequence tagging models such as GECToR (Omelianchuk et al., 2020) predict token-level edits using pre-trained encoders. Seq2seq systems have progressed from RNNs (Yuan and Briscoe, 2016) and CNNs (Chollampatt and Ng, 2018) to Transformer-based architectures like T5 (Rothe et al., 2021). More recently, LLMs generate corrections through conditional text generation, leveraging broad pre-training knowledge. Few or zero-shot GEC prompting of LLMs (e.g., GPT-4, LLaMA-2) achieves relatively low ERRANT scores due to their tendency to rewrite rather than minimally edit the texts, meaning that they drift from the reference edits and over-correct (Davis et al., 2024; Omelianchuk et al., 2024). However, none of these state-of-the-art GEC systems have been evaluated on MWE errors, due to the lack of MWE-annotated GEC data and automated error-type evaluation tools.

In this paper, we work with essays written by learners of English using the Write & Improve web application¹ (Yannakoudakis et al., 2018). These essays have already been publicly released for research use as part of the BEA Shared Task 2019 (Bryant et al., 2019) and the Write & Improve Corpus 2024 (Nicholls et al., 2024). We engaged expert annotators to mark up multiword expressions in the texts, in order to rigorously analyse how learners of different proficiency levels use MWEs in their writing. This also provides researchers with a new and sizeable MWE-annotated dataset. The annotation is then used to investigate how automatic error correction systems perform in detecting and correcting MWE errors. The main **contributions** of this work are as follows:

1. *We provide the first publicly available dataset of learner English essays annotated with MWEs.* We show statistics on how frequently learners use MWEs, how often they use them correctly, and what type of MWEs they use. This will benefit the research community to assess learners' understanding of MWEs and facilitate the development of better GEC and

MWE identification models.²

2. *We take the first steps towards providing MWE-informed feedback.* We introduce MWE error edits to an error annotation tool. Annotated MWE learner data will enable us to provide more informative feedback as opposed to purely token-level feedback which does not account for attempted use of MWEs.³
3. *We report the performance of state-of-the-art GEC systems on MWE error edits.* By developing an MWE identification system on our data, we enable investigating GEC models for their performance on MWE spans. We present a case study on how the MWE labels can be used in an edit-based GEC model and improve the GEC performance on MWE error types.

2 Multiword Expressions

Multiword expressions (MWEs) are combinations of two or more words that function as a single unit and often display idiosyncratic syntactic, semantic, or lexical behaviour (Baldwin and Kim, 2010). Despite comprising individual words, their meanings and grammatical properties are not always predictable from their components or standard grammatical rules. They include a wide variety of expressions such as idioms (e.g., *kick the bucket*), phrasal verbs (e.g., *give up*), light verb constructions (e.g., *take a look*), compound nouns (e.g., *traffic jam*), and others. These expressions are pervasive in natural language and play a crucial role in fluent and native-like language use.

Some defining properties of MWEs include:

- **Semantic non-compositionality:** The meaning of the whole expression cannot always be deduced from the meanings of its parts (e.g., *spill the beans* meaning *reveal a secret*).
- **Lexical rigidity:** MWEs often resist substitution, modification, or inflection (e.g., *by and large* cannot be changed to *by and huge*).
- **Syntactic irregularity:** MWEs can exhibit non-standard or fixed syntactic structures that defy general grammatical rules (e.g., *take place* cannot be passive like in *place is taken*.)

²The dataset is publicly available at: <https://researchdatasets.cambridge.org/datasets/mwe-gec-annotation>

³Our scripts are available at: <https://github.com/shivaat/mwe-gec>

¹<https://writeandimprove.com/>

- Frequency and conventionality: MWEs are fixed or semi-fixed expressions that are frequent and widely recognised within a language community.

Because of their irregularities and unpredictability, MWEs pose significant challenges for both language learners (Arnon and Christiansen, 2017) and computational language processing systems (Shwartz and Dagan, 2019) in terms of accurate usage and comprehension.

There is a lack of consensus in the literature on the precise definition and scope of MWEs, and scholars differ in their criteria for inclusion. Tsvetkov and Wintner (2014) define MWEs operationally as expressions which have to be stored in the lexicon of some NLP application. Motivated by this, we consider MWEs to be non-compositional expressions whose presence in a lexicon would be beneficial for language learners. Note that no dictionary is comprehensive enough to list all MWEs. Therefore, automatic approaches are required to help explore them. Automatic models rely on datasets to learn patterns and make accurate predictions. High-quality annotated datasets are essential for evaluating automatic models (Ide et al., 2025).

3 MWE-annotated Dataset

We obtained a sample of 1,850 learner English essays from two existing corpora derived from Write & Improve user submissions (Yannakoudakis et al., 2018). The sample comes from the dataset released for the BEA Shared Task 2019 on GEC in English (Bryant et al., 2019), as well as the more recent Write & Improve Corpus 2024 (Nicholls et al., 2024). The essays have already been corrected for grammatical errors by experienced, native-speaker annotators, providing high-quality reference corrections for reliable system evaluation.

Working with expert annotators, we added MWE annotations to the existing GEC annotations. This way, we can exploit the combined sets of annotations to investigate learners' use of MWEs and their errors. MWE-annotated learner data enables us to improve GEC and provide more informative feedback. We annotate the human-corrected side of the essays so annotators do not have to struggle with identifying ungrammatical MWEs. This also facilitates the analysis of learners' knowledge of MWEs (whether they use them correctly or not).

Annotation guidelines We prepared a comprehensive set of guidelines for annotating MWEs

(focusing on non-compositionality). We follow Schneider et al. (2014) and Savary et al. (2017) both for annotation guidelines and tagging schemes which involve sequence tagging while allowing for discontinuous MWEs. We annotate MWE types including compound nouns, propositional phrases, verb particle constructions, idioms, light verb constructions, complex propositions, complex subordinate, discourse connectives, fixed phrases, and binomials (i.e., pairs of words linked by conjunctions: e.g., *safe and sound*). In our annotation guidelines we pay specific attention to the Cambridge Advanced Learner's Dictionary (CALD)⁴ and the MWEs listed there. We consider it important to be able to map MWEs to CALD in future, in order to give better learning feedback.

Data selection We first selected the development set of the BEA-2019 data (Bryant et al., 2019). One of the advantages of annotating this data is that it is already in the public domain as a benchmark for GEC, so performance of GEC systems can be compared against each other. Another advantage of this dataset is that it is roughly evenly distributed across CEFR levels (A: beginner, B: intermediate, C: advanced; Council of Europe, 2001) plus native data, so we will be able to investigate MWE usages across different proficiency levels. This set contains 350 documents/essays which consist of 4.4k sentences and 87k words/tokens. We also selected a sample of 750 essays from the BEA-2019 training data and a sample of 750 essays from the Write & Improve Corpus 2024 (Nicholls et al., 2024) (evenly distributed across CEFR levels). This is to ensure broad coverage of different types of expressions over a span of time.

Annotation process We first performed a pilot study on a small number of essays from the BEA-2019 training set in order to calibrate the human annotators. The two annotators were native speakers of English and experts in language teaching. Disagreements between the two annotators were reviewed by a third expert lexicographer, followed by discussions and improvements to the annotation guidelines. We provided annotators with word and sentence tokenised texts to make sure the MWE annotations were consistent with the existing error correction annotations. The annotation platform allowed the annotators to not only highlight

⁴<https://dictionary.cambridge.org/dictionary/english/>

original	corrected
<i>In the other side , they might be wanted to safe-guard national image .</i>	<i>On_B the_I other_I hand_I , they might want to safeguard the national image .</i>
<i>Follow reading to know why ! .</i>	<i>Carry_B on_I reading to know why !</i>
<i>All of us spend a good time .</i>	<i>All of us have_B a good_I time_I .</i>
<i>Sometimes the employers do strike and we clients do not known in time as a result we can not get on time to work .</i>	<i>Sometimes the staff go_B on_I strike_I and we customers do not know in_B time_I . As_B a_I result_I , we can not get to work on_B time_I .</i>

Table 1: Examples from the dataset of original learner essay sentences (left) and their corrected versions (right).

sequences of words as MWEs, but also exclude arbitrary words in between the main components of an MWE (e.g., exclude *it* in *give it up*). The 350 corrected essays from the BEA-2019 dev data were annotated by both annotators. Double annotation facilitated a consistency check and helped measure inter-annotator agreement.

Inter-annotator agreement We measured inter-annotator agreement using F_{span} , defined in Ramisch et al. (2018) as an optimistic estimator which ignores chance agreement.⁵ It computes the F1 performance of one annotation in detecting exact spans of MWEs as if it is supposed to predict the other. The agreement between the two annotators on this MWE annotation task was $F_{span} = 0.896$. This level of agreement is strong compared to the same measure reported for datasets in Ramisch et al. (2018). Finally, the two sets of training data (750 essays in each) were split between the two annotators for efficiency and to work within pre-determined time constraints.

Annotation results Table 1 shows example sentences from the dataset where the tags *_B* and *_I* attached to the tokens represent the beginning and the continuation of an MWE, respectively.⁶ Annotators tagged a total of 2,973 MWEs in dev and 9,541 in train. More details on the statistics of the dataset are included in Appendix A. Having obtained these MWE annotations, we were then able to combine them with GEC error annotations in order to identify which edits are involved in MWEs. Next section details how we determine and use the

⁵We could not compute more common metrics such as *Kappa* because these metrics are ill-suited to this kind of token-span tagging task, and it is difficult to have an optimal estimation of observed and expected agreements.

⁶Note that the arbitrary words in between MWE component words, i.e. the words that can be replaced with other words such as *a* in *have a good time* are not tagged.

correlations between MWEs and GEC error edits.

4 MWE-informed GEC

4.1 Introducing MWEs to ERRANT

The original error annotations of the dataset capture various types of grammatical and lexical errors that are categorised using the ERRANT framework (Bryant et al., 2017). ERRANT standardises error types across systems by extracting edits from parallel original and corrected sentences, and classifies them using a rule-based error type framework based on POS tags and lemmas. It includes 25 main error categories, which are further classified as ‘Missing’, ‘Unnecessary’, or ‘Replacement’ edit types (i.e., the correction involves insertion, deletion or substitution, respectively).

In order for ERRANT to adopt MWE errors as a new error type we write scripts to find overlaps between already detected ERRANT error spans and MWE spans. Then we expand the ERRANT error spans to cover the identified MWEs. For example, if in the text *breathing pure air* (corrected to *breathing fresh air*), ERRANT tagged *pure* as R:ADJ (replacement adjective), we modify it to include *pure air* as an R:MWE (replacement MWE) error.⁷ Another example is given below in M²-format – a U:DET edit which indicates an unnecessary determiner. After our processing, the scope and type of the edit are expanded to an MWE replacement (R:MWE) involving the correction of the original sequence *for the better or worse* to *for better or worse*.

⁷We exclude spelling errors, noun number errors, subject-verb agreement errors, and verb tense errors, when introducing the modifications, because we believe these are not semantic errors related to learners’ knowledge of MWEs.

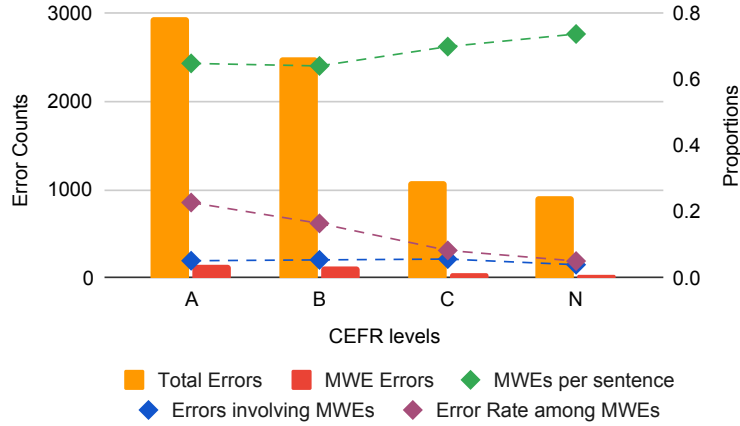


Figure 2: Distribution of MWEs and error statistics across CEFR levels {A,B,C} and native speakers (N).

S It is up to the average person to decide if this alteration will be for the better or worse .

A 15 16|||U:DET|||||REQUIRED|||-NONE-|||0

S It is up to the average person to decide if this alteration will be for the better or worse .

A 14 19|||R:MWE|||for better or worse|||REQUIRED|||-NONE-|||0

This modification helps us compare the rate of MWE errors compared to other ERRANT error types (see Section 5). We can also compare how different automatic error correction systems work when it comes to MWE errors. Another advantage of the addition of this MWE error type is that a model that follows this framework for GEC may not only correct the error, but also has pedagogical potential, in that it can be used to give feedback to writers by highlighting the multiword expressions that the writer should learn.

4.2 Case study: EditScorer

In order to prove the effectiveness of our dataset for current GEC systems, we focus on EditScorer (Sorokin, 2022), which separates error correction into two stages: 1) generation of candidate edits, and 2) ranking those edits to choose the best correction. This enables fine-grained control over the correction process and can be integrated with any edit generation system. Because it operates on elementary edits and assigns a score to each proposed correction, it offers a more interpretable correction process than end-to-end generation models. In this work, we focus on the second step which is ranking the edits and is more straightforward. Assuming

that we have the MWE edits generated (for example in a rule-based scenario), we aim to show how the ranking system can benefit from fine-tuning on our dataset.

In the original EditScorer, the ranker is trained on GECToR-generated edits (Omelianchuk et al., 2020). Using our dataset, we add the gold MWE edits (assuming they are generated) to the candidate edits and contrary to the original EditScorer, we keep the original edit spans that might contain ‘*k*-to-*l*’ edits with either $k > 1$ or $l > 1$. The re-ranking module in EditScorer is a RoBERTa-based transformer (Liu et al., 2019) which we fine-tune for 5 epochs. We compare the performance of the original EditScorer to the one fine-tuned on our dataset with MWE edits (see Section 6.2).

5 Dataset Characteristics and Analysis

After adding MWE errors to ERRANT (as explained in Section 4.1), we study the MWE errors in the dev data and make the following observations which demonstrate the initial advantages of adding MWE errors:

- 13.1% of all MWEs written by learners contain an error.
- MWE errors constitute 5% of all errors in the data, and they are as frequent as Noun and Verb errors that are among the most frequent and important errors when it comes to GEC.
- MWE errors account for 6.8% of the edits that ERRANT was unable to confidently classify (tagged OTHER).
- Among the most frequent erroneous MWEs in our dataset are the expressions *a lot of*, *on the other hand*, and *get along*. Learners struggle

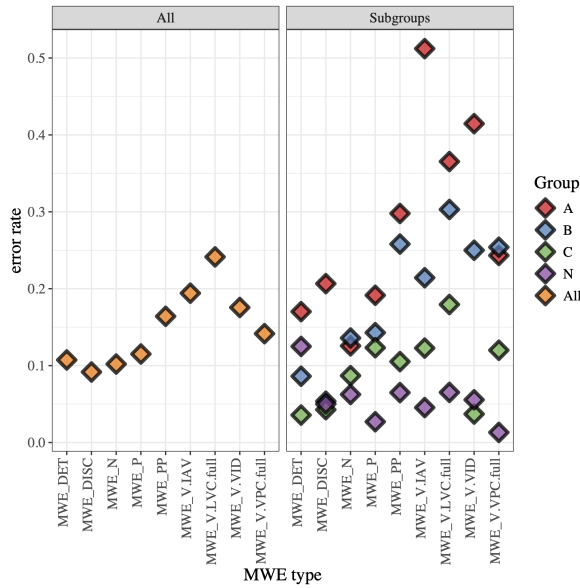


Figure 3: MWE error rates by MWE type for CEFR levels {A,B,C} and native speakers (N).

with the correct use of articles, or choosing the correct light verbs.

5.1 Errors across CEFR levels

Our dataset contains learner essays across three CEFR levels (A, B, and C) plus essays written by native speakers (N). The distribution of errors and MWEs across these four different groups is shown in Figure 2. In this figure, we see that the proportion of MWEs per sentence steadily increases with proficiency (green diamonds), suggesting that higher-level learners and native speakers use MWEs more frequently. The error rate for MWE usage (purple diamonds) shows that higher level learners tend to make fewer mistakes in the MWEs they use. The total error counts (orange bars) thus decrease with proficiency, as do MWE-related errors. However, the percentage of MWE errors relative to total errors (blue diamonds for errors involving MWEs) remains fairly stable across learner levels (around 4–5%), indicating that MWEs consistently pose challenges for learners, regardless of overall proficiency. This means that no matter the proficiency of a learner, MWE errors are an important error type to be corrected.

5.2 Automated classification of MWE categories

Exploring different types of MWEs in relation to learner errors can reveal nuanced aspects of second language acquisition. In this section, we present an automatic classification of MWEs into different

categories derived from the STREUSLE annotation scheme (Schneider et al., 2014; Liu et al., 2021).

The MWE tags contain lexical category labels from STREUSLE (see Table 3 in Appendix A in Liu et al. (2021) for the list of the tags, also added to Appendix B of this paper). Most of the tags are defined in line with Universal POS tag definitions. They also contain verbal MWE tags following Savary et al. (2017). The final number of MWE types is 22 in STREUSLE train and dev data.

We use a combination of STREUSLE and the Parseme Corpus v.1.3 (Savary et al., 2023) for training. In particular, we keep the STREUSLE test set as a development set, and our MWE annotated BEA 2019 dev set as a blind test set. Then we use sentences with MWEs from all other data as a training set (i.e., STREUSLE train/dev + Parseme train/dev/test). As a result we have 3,819 MWEs in the train and 283 MWEs in the dev data.

We fine-tune the pre-trained transformer-based model DeBERTa-v3-large (He et al., 2023) for our text classification task. DeBERTa is an encoder-only language representation model leveraging disentangled attention mechanisms and enhanced mask decoding. The fine-tuning process involved adapting the model’s pretrained weights to our domain-specific dataset by updating all parameters, including the transformer encoder and the classification head, using the cross-entropy loss function.

Following Tayyar Madabushi et al. (2022), for each MWE instance, we combine the context sentence with the MWE sequence as a data sample to be tokenised and encoded by the model.⁸ We use their baseline classification setup. Our target output is what MWE category the MWE instance in the context sentence belongs to. During the development phase we find the optimal hyperparameters to be learning rate $2e^{-5}$, batch size 32, warm-up steps 50, and we get the best F1-score after 8 epochs. The classifier achieves 90.8% accuracy and a macro F1 score of 76.8% on the development data, demonstrating solid overall performance.

After classifying MWEs into different types, we depict our observations on MWE error rates in Figure 3. Among the 18 MWE types that are predicted in our data, 9 have more than 10 occurrences in the dev set: MWE_DET, MWE_DISC, MWE_N, MWE_P, MWE_PP, MWE_V.IAV, MWE_V.LVC.full, MWE_V.VID, MWE_V.VPC.full.

⁸We only include the main component words of a MWE, excluding the arbitrary words in between for the classification.

	STREUSLE Test LinkAvg F_1	CoAM Test MWE-based F_1	Our GEC-MWE data MWE-based F_1
Liu et al. (2021)	72.0	-	-
Ide et al. (2025)	-	57.8	-
DeBERTa-based MWEI (ours)	76.8	64.7	76.3

Table 2: MWE identification performance on STREUSLE, CoAM and our dataset.

and MWE_V.VPC.full. We show the error rate for these specific MWE types for different proficiency levels as well as all levels in Figure 3. We can see in the figure that:

- LVC.full is the most common MWE error type among learners, followed by IAV, with learners making a mistake in each category 24% and 19% of the time respectively.
- Consistent with the observations in Figure 2, less proficient learners make more MWE mistakes. In Figure 3, we further see that learners of lower levels (A and B) find verbal MWEs particularly challenging, while learners of higher level (C) and native speakers (N) are more consistent in terms of the type of MWE errors they make.
- Differences across proficiency levels suggest that errors in specific MWE types may serve as stronger indicators of proficiency, warranting targeted error correction strategies for each type.

6 Experiments

In this section, we evaluate the proposed approach in two stages. First, the MWE identification task takes a text as input and predicts the MWEs it contains, including their spans. We then use the MWE identification system to support grammatical error correction for MWE error edits. The following subsections report the results for MWE identification and for the evaluation of GEC performance on MWE error edits, respectively.

6.1 MWE Identification

In order to detect MWEs on any corrected learner text, we use a modified version of the MWE identification system, MTLB-STRUCT, from Taslimipoor et al. (2020), which is reported to still be the state-of-the-art in tagging MWEs according to the PARSEME shared task on identification and paraphrasing of MWEs (Scholivet et al., 2026). Our modified model is a transformer-based token classifier fine-tuned using DeBERTa-v3 (He et al., 2023) similar to Rossini and van der Plas (2026).

We train this system on two benchmark datasets – STREUSLE (Schneider et al., 2014) and CoAM (Ide et al., 2025) – to show its performance compared to previous work.

The results can be seen in Table 2. For STREUSLE, we perform evaluation using LinkAvg F_1 , as in Liu et al. (2021), while for CoAM we use MWE-based F_1 as used by Ide et al. (2025) for direct comparison. As can be seen in the table, our model outperforms state-of-the-art MWE identification systems on both datasets. We then fine-tune this system on our dataset and report the MWE-based F_1 performance in Table 2.⁹ This system is used to extract MWEs from the output of GEC systems in the following section in order to have a fine-grained evaluation for MWE error types.

6.2 GEC for MWE error edits

We explore a selection of recent GEC systems reported and referred to in Omelianchuk et al. (2024). The systems are: T5-11B (a fine-tuned transformer-based sequence-to-sequence model), Chat-LLaMA-2 (a fine-tuned instruction following LLM prompting model), GECToR-2024 (an edit-based GEC system where transformer-based encoder-only models generate corrective edits), and EditScorer (another edit-based system which splits GEC into two steps: generating and scoring edits, Sorokin, 2022). Furthermore, we focus on EditScorer (ES) as a case study and we calculate its performance in re-ranking the candidate edits given gold MWE edits (as explained in section 4.2). At first we look at ES with the original re-ranker (trained on GECToR as in the original EditScorer paper), and then we show our adapted ES where the re-ranker is trained including MWE-annotated edits.

We use our MWE identification system (as described in subsection 6.1) to tag MWEs on the corrected files. Then we add predicted MWEs to ER-RANT alignment, as described in Section 4.1. We look at the performance of GEC systems on the five most frequent MWEs: noun compounds (MWE-N), full light verb constructions (MWE-LVC.full),

⁹More details on the MWE identification experiments can be found in Appendix C.

Systems	Overall	MWE-N	MWE-LVC .full	MWE-IAV	MWE-VID	MWE-VPC .full
GECToR-2024	55.49	23.62	38.22	43.1	48.39	35.71
Chat-LLaMa-2	55.53	44.69	52.49	46.88	47.79	31.25
T5-11B	58.08	51.72	51.35	47.79	49.34	37.5
EditScorer (ES)	56.46	17.39	42.64	50.0	48.91	41.67
ES – original re-ranker	58.75	48.39	41.38	58.33	51.72	57.29
ES – re-ranker trained on MWE edits	56.52	68.09	54.91	69.85	58.82	62.5

Table 3: GEC performance ($F_{0.5}$) on the five most frequent MWE error types.

inherently adpositional verbs (MWE-IAV), verbal idioms (MWE-VID), and full verb particle constructions (MWE-VPC.full). Table 3 shows GEC performance in terms of $F_{0.5}$ scores.

Looking at the performance for different types of MWE errors in the right-hand columns of Table 3, we see that different systems find VPCs particularly challenging to correct. This may be because light verbs can group with many different particles to build different constructions with different meanings. It is surprising to see that different systems work very differently when it comes to nominal MWEs (e.g., noun compounds). The performance of the selected systems are very different for Noun-based MWEs with GECToR-2024 and EditScorer finding MWE-N particularly challenging.

While the first half of Table 3 provides a benchmark for the state-of-the-art system performance for MWE edits, in the second half we show the potential of our dataset. It is interesting to see that the original EditScorer (ES – original re-ranker) performs much better if it sees the MWE edits generated even if it is not trained on those types of edits. We see a particularly good improvement in the case of MWE-N which the original EditScorer found challenging. Furthermore, we see that ES with our re-ranker trained on MWE edits significantly improves the GEC results for different types of MWEs (scores shown in bold).

6.3 Ablation study

We conducted an ablation study to assess the contribution of each MWE type to our EditScorer (ES – re-ranker trained on MWE edits). The original system is trained with all MWE edit types, alongside standard edit tags. We retrained the system while removing one MWE category at a time and evaluated its performance on all five types.

Removing a given MWE category generally reduced the $F_{0.5}$ score for that category and often affected the scores of other MWE categories as well. These results suggest that the different MWE cat-

egories contribute jointly to the re-ranking model. The exception is MWE-VPC.full, where the removal of some other types slightly improves GEC performance for MWE-VPC.full. It is worth noting that in all cases the overall performance decreases slightly by removing any MWE category from training.

6.4 Impact of MWE Identification Errors on GEC Evaluation

The evaluation of MWE-specific GEC performance described above relies on automatically identified MWE information extracted from the system outputs. Consequently, errors made by the MWE identification system may affect the attribution of corrections to MWE-related error types and may lead to an inaccurate estimate of GEC performance. To assess the impact of this dependency, we conducted a small human-annotation study on the corrected texts produced by the best performing system (ES – re-ranker trained on MWE edits). The annotator manually annotated the MWE spans in the output. Comparing the manually tagged MWE spans with those identified automatically yielded an F_1 score of 74.8 for the MWE identification system (comparable to the performance reported for our MWE identification system in Table 2), suggesting that its behaviour on GEC-generated text is broadly consistent with its performance on the evaluation data.

We then repeated the MWE-specific GEC evaluation using the human-annotated MWE tags instead of the automatically predicted tags. The resulting $F_{0.5}$ scores were 77.9 for MWE-N, 60.9 for MWE-LVC.full, 79.3 for MWE-IAV, 70.5 for MWE-VID, and 64.4 for MWE-VPC.full. These scores are higher than those obtained using automatic MWE identification for all MWE error categories (see the last row of Table 3). This indicates that errors in the automatic identification stage lead to an underestimation of MWE-specific GEC performance. It also suggests that improvements in MWE identification could yield more accurate estimates of GEC

Excluded MWE type	Overall	MWE-N	MWE-LVC	MWE-IAV	MWE-VID	MWE-VPC
None	56.52	68.09	54.91	69.85	58.82	62.5
MWE-N	56.04	45.82	45.45	46.88	43.1	67.31
MWE-LVC	56.18	37.04	50.39	55.15	41.67	64.66
MWE-IAV	56.32	37.85	57.05	48.39	42.97	64.66
MWE-VID	56.26	37.04	58.82	55.15	44.35	60.34
MWE-VPC	55.74	42.47	55.56	46.88	41.67	58.04

Table 4: Impact of MWE-type ablations on the performance of ES – re-ranker trained on MWE edits. Each row represents results where one MWE category is excluded when training the re-ranker.

performance, although the present analysis does not by itself establish that they would improve the GEC system’s actual correction ability.

7 Discussion

We provide researchers with the first learner English dataset annotated for multiword expressions. Our dataset, with 15.3k sentences in the training set and 4.4k sentences in the development set, is larger than similar counterparts: STREUSLE (Schneider et al., 2014) with 3.8k sentences, PARSEME 1.3 English (Savary et al., 2023) with 7.4k sentences, and CoAM (Ide et al., 2025) with 1.3k sentences. Our study leverages previous work, in particular STREUSLE and PARSEME, for preparing annotation guidelines and also for MWE type classification. CoAM represents some of the most recent work on MWE tagging. It includes automatic MWE type classification. However, it is small and also follows the old DiMSUM for MWE type tagging (Schneider et al., 2016); STREUSLE can be considered an upgraded version of DiMSUM. Our dataset can be used both for building MWE identification systems and also evaluating and improving GEC systems.

Classifying MWEs into different types is beneficial as we can see in Section 5.2: learners perform differently for different types of expressions. This is even more interesting when looking at the performance of learners of different proficiency levels. In the case of idioms, learners may misinterpret the meaning of MWEs, sometimes translating them too literally from their L1 or using them in inappropriate contexts. In the case of light verb constructions, learners may struggle to select the correct light verb (*do a decision* instead of *make a decision*), or they may omit the article (*make decision*). These errors are subtle and may not disrupt clarity in context but they do indicate developmental stages. GEC and feedback generation systems would benefit from being MWE-informed in order to provide more pedagogically-sound feedback.

The MWE-annotated version of the BEA 2019 dev dataset enables researchers to evaluate their GEC and feedback generation models further on MWE errors as we did for a selection of four SOTA systems. Ide et al. (2025) utilised instruction-tuned large language models for MWE identification, but found that both LLM-based approaches and models relying on external resources such as WordNet exhibited low recall rates. We performed standard Transformer-based token classification with BIO-style labelling for our sequence tagging rather than using LLM prompting, which has not yet shown promising results in MWE identification (Rossini and van der Plas, 2026; Arslan et al., 2025; Hashiloni et al., 2025). It is worth noting that LLM-based models that focus on generating texts are not recommended for GEC either (Omelianchuk et al., 2020) because their tendency to over-correct is not necessarily beneficial for language learning purposes (Bryant et al., 2023; Davis et al., 2024).

8 Conclusion

We release the first dataset that is tagged for both grammatical errors and MWEs. We have added MWE annotations to a sample of existing grammatical error correction data in English. We performed extensive analysis of different types of MWEs used (in)correctly by learners of different proficiency levels, thereby revealing which MWEs are more challenging at different learner levels. We proposed an approach to introduce MWE error edits to ER-RANT (a standard error annotation toolkit) and presented a case study where using the new labels can improve GEC for MWEs, which is potentially beneficial for pedagogical purposes. Our dataset can be used to further evaluate GEC systems and we provide a benchmark of how SOTA models perform on different types of MWEs. Future work can include research into how MWE usage develops in learning a second language and educational applications with MWE-informed assessment and feedback generation.

Limitations

Annotators were able to tag embedded MWEs such as *a great deal in gave her a great deal of confidence*. However, the .json files extracted from the annotation platform, at the moment, do not always extract the tags properly and consider them as a gap for the outer MWE. Also, when the inner MWE is part of the outer MWE – such as *year off in take a year off* – it is not clear what is the ending of which expression in the .json files extracted from the annotation platform. These cases are very rare in the dataset and we have the tags for outer MWEs in the dataset for such cases. Furthermore, ERRANT edits are continuous spans of texts. Even though our MWE annotation using BIO labelling excludes arbitrary words from the tags (e.g. *great in take a great care of*), we do not yet have a way to exclude them when we introduce MWE errors to ERRANT.

Acknowledgments

The ALTA Institute is supported by Cambridge University Press & Assessment. We thank Gloria George and David Barnett from the ARC Annotation team at CUP&A for their work on this dataset.

References

- Inbal Arnon and Morten H. Christiansen. 2017. [The role of multiword building blocks in explaining 11–12 differences](#). *Topics in Cognitive Science*, 9(3):621–636.
- Doğukan Arslan, Hüseyin Anıl Çakmak, Gulsen Eryigit, and Joakim Nivre. 2025. [Using LLMs to advance idiom corpus construction](#). In *Proceedings of the 21st Workshop on Multiword Expressions (MWE 2025)*, pages 21–31, Albuquerque, New Mexico, U.S.A. Association for Computational Linguistics.
- Timothy Baldwin and Su Nam Kim. 2010. Multiword expressions. In *Handbook of Natural Language Processing, second edition.*, pages 267–292. CRC Press.
- Frank Boers and Seth Lindstromberg, editors. 2008. *Cognitive Linguistic Approaches to Teaching Vocabulary and Phraseology*. De Gruyter Mouton, Berlin, New York.
- Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. 2019. [The BEA-2019 shared task on grammatical error correction](#). In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–75, Florence, Italy. Association for Computational Linguistics.
- Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. [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)*, pages 793–805, Vancouver, Canada. Association for Computational Linguistics.
- Christopher Bryant, Zheng Yuan, Muhammad Reza Qorib, Hannan Cao, Hwee Tou Ng, and Ted Briscoe. 2023. [Grammatical error correction: A survey of the state of the art](#). *Computational Linguistics*, 49(3):643–701.
- Shamil Chollampatt and Hwee Tou Ng. 2018. [Neural quality estimation of grammatical error correction](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2528–2539, Brussels, Belgium. Association for Computational Linguistics.
- Council of Europe. 2001. *Common European Framework of Reference for Languages: learning, teaching, assessment*. Cambridge University Press, Cambridge.
- Daniel Dahlmeier and Hwee Tou Ng. 2012. [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*, pages 568–572, Montréal, Canada. Association for Computational Linguistics.
- Christopher Davis, Andrew Caines, Øistein E. Andersen, Shiva Taslimipour, Helen Yannakoudakis, Zheng Yuan, Christopher Bryant, Marek Rei, and Paula Buttery. 2024. [Prompting open-source and commercial language models for grammatical error correction of English learner text](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11952–11967, Bangkok, Thailand. Association for Computational Linguistics.
- Kai Golan Hashiloni, Ofri Hefetz, and Kfir Bar. 2025. [Easy as PIE? identifying multi-word expressions with LLMs](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 23771–23790, Suzhou, China. Association for Computational Linguistics.
- Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. [Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing](#). *Preprint*, arXiv:2111.09543.
- Yusuke Ide, Joshua Tanner, Adam Nohejl, Jacob Hoffman, Justin Vasselli, Hidetaka Kamigaito, and Taro Watanabe. 2025. [CoAM: Corpus of all-type multi-word expressions](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 27004–27021, Vienna, Austria. Association for Computational Linguistics.
- Nelson F. Liu, Daniel Hershcovich, Michael Kranzlein, and Nathan Schneider. 2021. [Lexical semantic recognition](#). In *Proceedings of the 17th Workshop on Multi-*

- word Expressions (MWE 2021)*, pages 49–56, Online. Association for Computational Linguistics.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. [RoBERTa: A robustly optimized BERT pretraining approach](#). *CoRR*, abs/1907.11692.
- Fanny Meunier and Sylviane Granger. 2008. *Phraseology in foreign language learning and teaching*. John Benjamins Publishing.
- Tomoya Mizumoto, Masato Mita, and Yuji Matsumoto. 2015. [Grammatical error correction considering multi-word expressions](#). In *Proceedings of the 2nd Workshop on Natural Language Processing Techniques for Educational Applications*, pages 82–86, Beijing, China. Association for Computational Linguistics.
- Diane Nicholls, Andrew Caines, and Paula Buttery. 2024. *The Write & Improve Corpus 2024: Error-annotated and CEFR-labelled essays by learners of English*. Cambridge University Press & Assessment.
- Kostiantyn Omelianchuk, Vitaliy Atrasevych, Artem Chernodub, and Oleksandr Skurzhashnyi. 2020. [GECToR – grammatical error correction: Tag, not rewrite](#). In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 163–170, Seattle, WA, USA → Online. Association for Computational Linguistics.
- Kostiantyn Omelianchuk, Andrii Liubonko, Oleksandr Skurzhashnyi, Artem Chernodub, Oleksandr Kornienko, and Igor Samokhin. 2024. [Pillars of grammatical error correction: Comprehensive inspection of contemporary approaches in the era of large language models](#). In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 17–33, Mexico City, Mexico. Association for Computational Linguistics.
- Magali Paquot and Sylviane Granger. 2012. [Formulaic language in learner corpora](#). *Annual Review of Applied Linguistics*, 32:130–149.
- Carlos Ramisch, Silvio Ricardo Cordeiro, Agata Savary, Veronika Vincze, Verginica Barbu Mititelu, Archana Bhatia, Maja Buljan, Marie Candito, Polona Gantar, Voula Giouli, Tunga Güngör, Abdelati Hawwari, Uxoia Iñurrieta, Jolanta Kovalevskaitė, Simon Krek, Timm Lichte, Chaya Liebeskind, Johanna Monti, Carla Parra Escartín, Behrang QasemiZadeh, Renata Ramisch, Nathan Schneider, Ivelina Stoyanova, Ashwini Vaidya, and Abigail Walsh. 2018. [Edition 1.1 of the PARSEME shared task on automatic identification of verbal multiword expressions](#). In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 222–240, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
- Carlos Ramisch, Agata Savary, Bruno Guillaume, Jakub Waszczuk, Marie Candito, Ashwini Vaidya, Verginica Barbu Mititelu, Archana Bhatia, Uxoia Iñurrieta, Voula Giouli, Tunga Güngör, Menghan Jiang, Timm Lichte, Chaya Liebeskind, Johanna Monti, Renata Ramisch, Sara Stymne, Abigail Walsh, and Hongzhi Xu. 2020. [Edition 1.2 of the PARSEME shared task on semi-supervised identification of verbal multiword expressions](#). In *Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons*, pages 107–118, online. Association for Computational Linguistics.
- Diego Rossini and Lonneke van der Plas. 2026. [Binary token-level classification with deberta for all-type mwe identification: A lightweight approach with linguistic enhancement](#). *Preprint*, arXiv:2601.19360.
- Sascha Rothe, Jonathan Mallinson, Eric Malmi, Sebastian Krause, and Aliaksei Severyn. 2021. [A simple recipe for multilingual grammatical error correction](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 702–707, Online. Association for Computational Linguistics.
- Agata Savary, Cherifa Ben Khelil, Carlos Ramisch, Voula Giouli, Verginica Barbu Mititelu, Najet Hadj Mohamed, Cvetana Krstev, Chaya Liebeskind, Hongzhi Xu, Sara Stymne, Tunga Güngör, Thomas Pickard, Bruno Guillaume, Eduard Bejček, Archana Bhatia, Marie Candito, Polona Gantar, Uxoia Iñurrieta, Albert Gatt, Jolanta Kovalevskaitė, Timm Lichte, Nikola Ljubešić, Johanna Monti, Carla Parra Escartín, Mehrnough Shamsfard, Ivelina Stoyanova, Veronika Vincze, and Abigail Walsh. 2023. [PARSEME corpus release 1.3](#). In *Proceedings of the 19th Workshop on Multiword Expressions (MWE 2023)*, pages 24–35, Dubrovnik, Croatia. Association for Computational Linguistics.
- Agata Savary, Carlos Ramisch, Silvio Cordeiro, Federico Sangati, Veronika Vincze, Behrang QasemiZadeh, Marie Candito, Fabienne Cap, Voula Giouli, Ivelina Stoyanova, and Antoine Doucet. 2017. [The PARSEME shared task on automatic identification of verbal multiword expressions](#). In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 31–47, Valencia, Spain. Association for Computational Linguistics.
- Nathan Schneider, Dirk Hovy, Anders Johannsen, and Marine Carpuat. 2016. [SemEval-2016 task 10: Detecting minimal semantic units and their meanings \(DiMSUM\)](#). In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 546–559, San Diego, California. Association for Computational Linguistics.
- Nathan Schneider, Spencer Onuffer, Nora Kazour, Emily Danchik, Michael T. Mordowanec, Henrietta Conrad, and Noah A. Smith. 2014. [Comprehensive annotation of multiword expressions in a social web corpus](#). In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14)*, pages 455–461, Reykjavik, Iceland. European Language Resources Association (ELRA).

- Manon Scholivet, Agata Savary, Carlos Ramisch, Eric Bilinski, Takuya Nakamura, Maria Mitrofan, and Vasile Pais. 2026. [Edition 2.0 of the PARSEME shared task on multilingual identification and paraphrasing of multiword expressions](#). In *Proceedings of the 22nd Workshop on Multiword Expressions (MWE 2026)*, pages 254–275, Rabat, Morocco. Association for Computational Linguistics.
- Uliana Sentsova, Debora Ciminari, Josef Van Genabith, and Cristina España-Bonet. 2025. [MultiCoPIE: A multilingual corpus of potentially idiomatic expressions for cross-lingual PIE disambiguation](#). In *Proceedings of the 21st Workshop on Multiword Expressions (MWE 2025)*, pages 67–81, Albuquerque, New Mexico, U.S.A. Association for Computational Linguistics.
- Vered Shwartz and Ido Dagan. 2019. [Still a pain in the neck: Evaluating text representations on lexical composition](#). *Transactions of the Association for Computational Linguistics*, 7:403–419.
- Anna Siyanova-Chanturia and Ron Martinez. 2014. [The idiom principle revisited](#). *Applied Linguistics*, 36(5):549–569.
- Alexey Sorokin. 2022. [Improved grammatical error correction by ranking elementary edits](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11416–11429, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Shiva Taslimipoor, Sara Bahaadini, and Ekaterina Kochmar. 2020. [MTLB-STRUCT @parseme 2020: Capturing unseen multiword expressions using multi-task learning and pre-trained masked language models](#). In *Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons*, pages 142–148, online. Association for Computational Linguistics.
- Shiva Taslimipoor, Christopher Bryant, and Zheng Yuan. 2022. [Improving grammatical error correction for multiword expressions](#). In *Proceedings of the 18th Workshop on Multiword Expressions @LREC2022*, pages 9–15, Marseille, France. European Language Resources Association.
- Harish Tayyar Madabushi, Edward Gow-Smith, Marcos Garcia, Carolina Scarton, Marco Idiart, and Aline Villavicencio. 2022. [SemEval-2022 task 2: Multilingual idiomaticity detection and sentence embedding](#). In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 107–121, Seattle, United States. Association for Computational Linguistics.
- Yulia Tsvetkov and Shuly Wintner. 2014. [Identification of multiword expressions by combining multiple linguistic information sources](#). *Computational Linguistics*, 40(2):449–468.
- Helen Yannakoudakis, Øistein E Andersen, Ardeshtir Geranpayeh, Ted Briscoe, and Diane Nicholls. 2018. [Developing an automated writing placement system for ESL learners](#). *Applied Measurement in Education*, 31(3):251–267.
- Zheng Yuan and Ted Briscoe. 2016. [Grammatical error correction using neural machine translation](#). In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–386, San Diego, California. Association for Computational Linguistics.

A More Details on the Proposed Dataset

We re-annotated existing GEC-annotated data for MWEs following our comprehensive guidelines for different categories of MWEs. The exercise consisted of annotating MWEs in learners’ essays which have already been corrected by human experts. The annotators grouped tokens into MWEs. The annotation platform let them exclude arbitrary words in between the main components of a MWE where it is discontinuous.

We follow the BIO-style (Beginning, Inside, Outside) tagging scheme similar to STREUSLE (Schneider et al., 2014), where B indicates the token is the beginning of a new MWE, I indicates that the token is a continuation of an MWE, and O indicates that the token is not part of an MWE. Small *b*, small *i*, and small *o* are also used for nested MWEs. Using this tagging scheme, the expression *have a good time*, is presented as *have_B a good_I time_I*. The annotation platform also allows for nested MWEs, where for example in *gave_B me a_b lot_i of_i confidence_I* in the dataset, the nested expression is tagged with small *b* and *i*.

Table 5 details the statistics of the MWE-annotated dataset in our train and development sets. This dataset is also useful for MWE identification research. After finding the correlations between the MWEs and error spans as explained in Section 4.1, we study MWE errors accross CEFR levels in Section 5. The MWE error statistics corresponding to Figure 2 are presented in Table 6 for more clarifications.

B Different Categories for Automatic Identification of Multiword Expressions

For automatic identification of MWE categories in our dataset, we follow the STREUSLE tags (listed in Liu et al. (2021)). The tags identified in our dataset are described in Table 7. Most of the tags,

	essays	sents	tokens	MWEs	MWE errors
train	1,500	16k	303k	9,892	1,079
dev	350	4.4k	89k	2,973	388

Table 5: Statistics of the dataset.

	A	B	C	N
Total Errors	2953	2500	1090	929
MWE Errors	153	136	62	37
MWEs per sentence	0.648	0.640	0.699	0.737
Errors involving MWEs	0.052	0.054	0.057	0.040
Error Rate among MWEs	0.228	0.165	0.083	0.051

Table 6: MWE error rates (in dev data) across different proficiency levels.

except for the verbal MWEs, the nominal MWEs (N), idiomatic prepositional phrase MWEs (PP), and adposition MWEs (P), are in line with Universal POS tag definitions. In these cases the category of an MWE is determined by the part of speech to which its head word belongs.

After classifying MWEs into different categories, we studied error rates for different MWE types across CEFR levels in Section 5.2. The statistics corresponding to Figure 3 can be seen in Table 8 for more clarifications.

C The MWE Identification System Details

Our MWE identification system is a transformer-based token classifier similar to MTLB-STRUCT (Taslimipour et al., 2020) fine-tuned using DeBERTa-v3 (He et al., 2023). It predicts BIO labels, where B and I tags in our training datasets are followed by the type of MWEs (e.g. N, P, V.IAV, V.LVC.full, etc). We follow MTLB-STRUCT single-task setting for most part, and we find learning rate $3e^{-5}$, batch size 16, and number of epochs 8 as the best parameters during development.

As explained in Section 7, there are several small MWE-tagged datasets similar to ours. For our experiments in Section 6.1, we first combined STREUSLE (with 2730 MWEs in the training and development sets) and the Parseme-v3 English dataset (Savary et al., 2023) (with an additional 1,114 MWEs). The system fine-tuned on the combination of these two datasets is considered out-of-domain for our dataset. However, it can be directly compared to the BERT-based model by Liu et al. (2021) on the STREUSLE test set using MWE LinkAvg F_1 . In Table 9, we can see that our model outperforms that of Liu et al. (2021).

To improve MWE annotation for learner corpora (e.g. GEC datasets), we further fine-tune the MWE identification system on our MWE-annotated training data which is a sample of W&I data (including a sample from the BEA training set) annotated for MWEs as explained in Section 3. The MWEs in this data are type-classified using the method explained in Section 5.2. we report system performance on our MWE-annotated data using the MWE-based Precision (P), Recall (R), and F1 scores from the Parseme 2020 shared task (Ramisch et al., 2020). We can see the improvement in MWE identification on our dev data in Table 9. This system is used to identify MWEs in any error-corrected text.

D EditScorer Re-ranking

In this section, we report the performance of the ES re-ranker in more detail using Precision (P) and Recall (R) as well as $F_{0.5}$. The results are presented in Table 10. For almost all MWE types, the performance improvement achieved by our re-ranker, which was trained on MWE edits, is attributable to gains in both Precision and Recall. The exception is MWE-VPC.full, for which Precision decreases slightly, whereas Recall increases substantially.

MWE Category	Description	Examples
MWE_ADJ	Adjective	<i>upside down</i>
MWE_ADV	Adverbial	<i>so far</i>
MWE_AUX	Auxiliary verbs	<i>have yet to</i>
MWE_CCONJ	Coordinating conjunction	<i>as well as</i>
MWE_DET	Determiner	<i>a lot of</i>
MWE_DISC	Discourse/pragmatic expression	<i>on the other hand</i>
MWE_N	Nominal expressions	<i>public transport</i>
MWE_P	Single-word or compound adposition	<i>in order to</i>
MWE_PP	prepositional phrase MWE	<i>in to minds</i>
MWE_PRON	Non-possessive pronoun	<i>one day</i>
MWE_SCONJ	Subordinating conjunction	<i>even if</i>
MWE_V.IAV	Inherently reflexive verbs	<i>to help oneself to food</i>
MWE_V.LVC.cause	Verb noun constructions in which the verb adds a causative meaning to the noun.	<i>give control</i>
MWE_V.LVC.full	Verb noun constructions in which the verb has no semantic content.	<i>make a decision</i>
MWE_V.MVC	Constructions consisting of multiple verbs.	<i>let go</i>
MWE_V.VID	Verbal expressions that have fully idiomatic interpretations.	<i>go bananas</i>
MWE_V.VPC.full	Fully non-compositional verb particle constructions in which the particle totally changes the meaning of the verb.	<i>give up</i>
MWE_V.VPC.semi	Semi non-compositional verb particle constructions in which the particle adds a partly predictable but non-spatial meaning to the verb.	<i>eat up</i>

Table 7: Description and examples for the MWE Categories predicted in our data.

MWE types	A	B	C	N	All
MWE-DET	0.170	0.087	0.036	0.125	0.107
MWE-DISC	0.207	0.054	0.043	0.05	0.092
MWE-N	0.126	0.136	0.087	0.063	0.102
MWE-P	0.192	0.143	0.123	0.027	0.115
MWE-PP	0.298	0.258	0.105	0.065	0.164
MWE-V.IAV	0.512	0.214	0.123	0.046	0.194
MWE-V.LVC.full	0.365	0.303	0.180	0.065	0.241
MWE-V.VID	0.415	0.25	0.037	0.056	0.176
MWE-V.VPC.full	0.243	0.254	0.12	0.013	0.142
total	0.228	0.165	0.083	0.051	0.131

Table 8: Different MWE type errors for different proficiency levels.

	Our dev data			STREUSLE Test
	P	R	F_1	LinkAvg F_1
Liu et al. (2021)	-	-	-	72.0%
Out-of-domain training	1825/3668 = 0.498	1825/2973 = 0.614	55.0%	76.8%
In-domain training	2258/2949 = 0.766	2258/2973 = 0.756	76.3%	-

Table 9: MWE identification performance on our dev data and the STREUSLE test set.

EditScorer (ES)	Overall			MWE-N			MWE-LVC.full			MWE-IAV			MWE-VID			MWE-VPC.full		
	P	R	$F_{0.5}$	P	R	$F_{0.5}$	P	R	$F_{0.5}$	P	R	$F_{0.5}$	P	R	$F_{0.5}$	P	R	$F_{0.5}$
original re-ranker	69.0	36.9	58.75	75.0	20.0	48.39	50.0	24.5	41.38	70.0	35.0	58.33	60.0	33.3	51.72	68.8	34.4	57.29
re-ranker trained on MWE edits	64.8	37.4	56.52	80.0	42.7	68.09	61.3	38.8	54.91	79.2	47.5	69.85	64.0	44.4	58.82	65.4	53.1	62.5

Table 10: GEC performance (P, R, and $F_{0.5}$) for our two versions of ES from Table 3.