Improving Grammatical Error Correction for Multiword Expressions

Shiva Taslimipoor[1] **Christopher Bryant**[1] **Zheng Yuan**[2,1]
[1] ALTA Institute, Department of Computer Science and Technology, University of Cambridge, U.K.
{firstname.lastname}@cl.cam.ac.uk
[2] Department of Informatics, King's College London, U.K.
zheng.yuan@kcl.ac.uk

Abstract

Grammatical error correction (GEC) is the task of automatically correcting errors in text. It has mainly been developed to assist language learning, but can also be applied to native text. This paper reports on preliminary work in improving GEC for multiword expression (MWE) error correction. We propose two systems which incorporate MWE information in two different ways: one is a multi-encoder decoder system which encodes MWE tags in a second encoder, and the other is a BART pre-trained transformer-based system that encodes MWE representations using special tokens. We show improvements in correcting specific types of verbal MWEs based on a modified version of a standard GEC evaluation approach.

Keywords: Multiword Expressions, Grammatical Error Correction

1. Introduction

Second language learners make various kinds of errors in their writing. State-of-the-art Grammatical Error Correction (GEC) systems attempt to correct these errors primarily using neural machine translation technology (Yuan and Briscoe, 2016). These systems are often biased towards correcting the most common error types, however, such as determiner, preposition and spelling errors. Learners can nevertheless also be more creative in generating semantically incorrect phrases such as *by the other side* or *in the other hand* rather than *on the other hand*, or *dream becomes true* instead of *dream comes true*.

Multiword expressions (MWEs), which are combinations of two or more words with syntactic and semantic idiosyncratic behaviours (Sag et al., 2002), are known to be challenging for language learners (Christiansen and Arnon, 2017; Meunier and Granger, 2008). However, like most machine translation (MT) systems, current GEC systems do not take them into consideration. One important challenge involved in the natural language understanding of these expressions is that their meaning deviates from the meaning of their constituent words. Shwartz and Dagan (2019) show that even the state-of-the-art contextualised word representation models have problems in detecting such meaning shifts and their performance is far from that of humans.

Previous studies pointed to the importance of MWEs in GEC. In particular, Mizumoto et al. (2015) merged the tokens in a MWE into a single unit and then applied phrase-based MT and reported a generally better performance for their GEC system that took MWEs into consideration. It has also been reported that such errors are related to learners' L1 (Nesselhauf, 2003). In line with this, Dahlmeier and Ng (2011) use L1-induced paraphrases to correct learners' erroneous use of collocations. Other works focusing on correcting collocation errors made by language learners include the studies by Kochmar (2016). She focused on adjective-noun and verb-object combinations and extracted the meaning representations of the combinations using models of compositional semantics in order to distinguish between the representations of the correct and incorrect content word combinations.

In this work, we deal with all types of MWEs which are difficult to correct based solely on standard contextualised information/embeddings. Specifically, we add MWE information to existing GEC systems in order to investigate how they can improve performance.

Contributions: We propose two different approaches to encode MWE information in existing GEC systems: 1) We augment an encoder-decoder transformer-based GEC model with a separate encoder which encodes MWE tags, and 2) We add special MWE tokens around automatically-detected MWEs in the input to help the encoder-decoder model learn a special representation for them. We show improvements in the performance of the two GEC systems especially in correcting specific types of verbal MWE errors.

2. Grammatical Error Correction

Most recent work on GEC treats the task as a monolingual machine translation problem from 'incorrect' to 'correct' English (Felice et al., 2014; Yuan and Briscoe, 2016; Grundkiewicz et al., 2019; Stahlberg and Kumar, 2021; Yuan et al., 2021). Specifically, given a corpus of parallel erroneous and grammatical sentences, the task is to generate corrected sentences from the erroneous sentences. Alternatively, another recent promising approach treats the task as a sequence labelling problem where each token label represents an edit in the sentence (e.g. KEEP, DELETE, REPLACE) (Awasthi et al., 2019; Omelianchuk et al., 2020; Stahlberg and Kumar, 2020).

Proceedings of the 18th Workshop on Multiword Expressions (MWE 2022) n@LREC2022, pages 9–15
Marseille, 25 June 2022. © European Language Resources Association (ELRA), licensed under CC-BY-NC-4.0

In this paper, we employ two different Transformer-based NMT systems (Vaswani et al., 2017) for GEC: 1) An encoder-decoder GEC system based on Yuan et al. (2021) and 2) A strong BART-based GEC system (Katsumata and Komachi, 2020). The main advantage of the first system is that it includes multi-encoders for representing different features. In particular, while Yuan et al. (2021) used the additional encoder to include features from grammatical error detection, we use the extra encoder to incorporate MWE tags into the model (see Section 4.2). In contrast, the main advantage of the second system is that it is a strong baseline GEC system that simply fine-tunes BART on in-domain GEC data (and hence does not rely on additional techniques such as re-ranking or ensembling) to produce results that are competitive with the state of the art. We add special tokens for MWEs to the data to allow the model to explicitly encode MWEs (Section 4.3).

3. Multiword Expression Identification

As far as we are aware, no dataset in GEC has been explicitly annotated with MWE information; i.e. we do not know which tokens comprise ungrammatical/grammatical MWEs in both the original and the corrected text. Since this information is necessary in an MWE-aware GEC system, we derive these labels automatically.

Our MWE identification system is a transformer-based pre-trained language representation model which we fine-tune for sequence tagging. The model is similar to MTLB-STRUCT (Taslimipoor et al., 2020) with the difference that we use ELECTRA rather than BERT and perform single-task learning for which they reported better performance than for multi-task training in most languages. ELECTRA (Clark et al., 2020) is a variation of BERT (Devlin et al., 2019) that is pre-trained to discriminate between original and replaced tokens rather than generate masked tokens, on the data. In order to predict various types of MWEs including noun compounds, e.g. *customer service*; set phrases, e.g. *as well*, *so far*; and idioms, e.g. *go the extra mile*, we fine-tune our system on a combination of the STREUSLE dataset (Schneider et al., 2014) and the English side of the PARSEME dataset (Ramisch et al., 2018). The newest version of STREUSLE, as used by Liu et al. (2021), contains more detailed/fine-grained tags for verbal MWEs (following Savary et al. (2017)). Both these datasets are tagged following a variation of IOB labeling (Inside, Outside, Beginning) where *O* indicates that the token is not part of an MWE, *B* indicates the token is the beginning of a new MWE and *I* indicates that the token is a continuation of an MWE. *B*, and *I* tags in these datasets are followed by the type of MWE.

For evaluating our MWE identification system, we follow Liu et al. (2021) and report standard STREUSLE evaluation metrics for MWEs and also PARSEME MWE-based metrics for verbal MWEs on the STREUSLE test set. Table 1 shows that our ELECTRA-based system outperforms the BERT-based system used by Liu et al. (2021).

	MWE LinkAvg			Verbal MWE-based		
	P	R	F_1	P	R	F_1
# Gold	433.5			66		
Liu et al. (2021)	82.0	64.3	72.0	-	-	63.9
Our system	**90.7**	**66.8**	**76.7**	**65.2**	**68.2**	**66.7**

Table 1: Overall performance of the MWE identification system on STREUSLE test set.

The MWE tags for English contain lexical category labels from STREUSLE including ADJ (adjective), ADV (adverb), DET (determiner) which are in line with Universal part-of-speech tags, AUX (auxiliary), DISC (discourse/pragmatic expression), N (noun, common or proper), P (single-word or compound adposition), PP (prepositional phrase MWE), and PRON (non-possessive pronoun, including indefinites like someone) which indicate the holistic grammatical status of strong multiword expressions plus the verbal MWE tags as follows:

- IAV (Inherently adpositional verbs, also called prepositional verbs e.g. *come accross*),
- LVC.full (light verb constructions in which the verb is semantically totally bleached, e.g. *make a decision*),
- VID (verbal expressions that have fully idiomatic interpretations, e.g., *go bananas*),
- VPC.full (fully non-compositional verb particle constructions, in which the particle totally changes the meaning of the verb, e.g. *give up*),
- VPC.semi (semi non-compositional verb particle constructions, in which the particle adds a partly predictable meaning to the verb, e.g. *eat up*)

Since verbal MWEs are often more challenging for learners (Siyanova and Schmitt, 2007), we particularly focus on this subset of MWE tags in our evaluation.

4. Experiments

4.1. MWE-Augmented GEC Data

Having built a system to annotate MWEs, we apply it to several popular GEC corpora, including the public FCE (Yannakoudakis et al., 2011), NUCLE (Dahlmeier et al., 2013) and W&I (Bryant et al., 2019). Specifically, we annotate the original, uncorrected side of the parallel data with MWE information and convert the annotations into two different formats for our experiments, as explained below.

4.2. Experiment 1: Using MWE in Multi-encoder GEC

Following the work of Yuan and Bryant (2021; Yuan et al. (2021), we incorporate additional MWE information into GEC by introducing a second encoder to

S	This	reminds	me	of	a	trip	that	I	have	been	to	.
3-class	O	O	O	O	O	O	O	O	O	B	I	O
23-class	O	O	O	O	O	O	O	O	O	B-V	I-V	O
T	This	reminds	me	of	a	trip	that	I	have	been	on	.

Table 2: An example sentence with MWE tags at different levels of granularity. 3-class: Begin, Inside, Outside; 23-class: Begin-Verb, Inside-Verb.

the standard Transformer encoder-decoder model. The original Transformer encoder reads the source sentence S_{src} and learns a vector representation c_{src} as before. An additional encoder is introduced to process any auxiliary MWE tags S_{mwe} and compute another representation c_{mwe} in parallel. The decoder now includes a new MWE multi-head attention which attends directly to the MWE encoder representation c_{mwe}, and a linear gating mechanism that combines the source multi-head attention and the new MWE multi-head attention.

A two-step training strategy is employed to train the new GEC model. In the first step, we follow the standard encoder-decoder model training procedure and train a sequence-to-sequence model on parallel Cambridge Learner Corpus (CLC) data (Nicholls, 2003) without MWE information using Fairseq (Ott et al., 2019). In the second step, we fine-tune this model using the auxiliary MWE-tagged data at different levels of granularity. Specifically, the model is fine-tuned on the MWE-tagged FCE, NUCLE and W&I training data where each token is tagged with a coarse (IOB) or fine-grained (IOB+type) MWE labels (Table 2).

4.3. Experiment 2: MWE Marker Tokens

Inspired by Baldini Soares et al. (2019) who used 'entity markers' as special tokens to mark the beginning and end of named entities, we similarly use special tokens to mark the spans of MWEs. This allows us to encode a representation of an MWE as a special unit. We augment our GEC training data with two reserved special tokens [MWE] and [/MWE] to mark the beginning and end of each MWE, respectively, as determined by the MWE identification system (Section 3), in both the original and corrected sides of the texts. We follow two scenarios for marking MWEs in parallel GEC data:

1. We predict MWEs in the *original* text and map the special tokens to the equivalent positions in the *corrected* text.
2. We predict MWEs in the *corrected* text and map the special tokens to the equivalent positions in the *original* text.

In the first case, we simply annotate the texts in the original (source) side with automatically identified MWE tags and use the GEC alignment algorithm ERRANT (Bryant et al., 2017) to automatically find the corresponding spans in the corrected (target) texts. This scenario represents the realistic use-case since we are always given the original text to be corrected. The disadvantage of this approach, however, is that the MWE identification system may not be very accurate since it is trained on native texts with no errors yet applied to ungrammatical text.

Model: Encoder-decoder	P	R	$F_{0.5}$
baseline	57.95	31.22	49.48
MWE-augmented [3-class]	57.80	33.60	50.53
MWE-augmented [23-class]	58.53	33.98	**51.14**

Table 3: Overall performance of the encoder-decoder GEC systems on BEA dev set.

Model: BART	P	R	$F_{0.5}$
baseline	56.08	37.73	51.11
MWE-augmented (1)	56.88	35.36	50.71
MWE-augmented (2)	57.21	36.71	**51.46**

Table 4: Overall performance of the BART GEC system fine-tuned on raw and MWE-tagged W&I data.

In the second case, we hence annotate MWEs in the corrected side, where they are more likely to be well-formed, and again find the equivalent spans in the original text using ERRANT. In contrast with the first case, the disadvantage of this second approach is that we do not have access to the corrected text in the realistic use-case even though the identified MWEs may be more reliable. We nevertheless explore this scenario for comparison with the first scenario. Figure 1 shows an example of a parallel sentence pair with marked MWEs. [1]

S: they also [MWE] made talks [/MWE] and presentations about the earth 's problems
T: they also [MWE] give talks [/MWE] and presentations about the earth 's problems

Figure 1: A sentence pair with marked MWEs.

In this experiment, we use a pre-trained BART model which we fine tune on the MWE-annotated W&I training corpus (Bryant et al., 2019). Katsumata and Komachi (2020) have shown that this model produces competitive results with the state of the art in GEC. Our addition of explicit MWE markers helps the model to better encode representations of MWEs.

4.4. Evaluation

We first report the general performance of our GEC systems with and without incorporating MWE tags in

[1] This example is the same in both scenarios, however, there are also cases where MWEs on one side are aligned with non-MWEs on the other side.

			Baseline GEC			MWE-augmented GEC		
	MWE type	#	P	R	$F_{0.5}$	P	R	$F_{0.5}$
Encoder-decoder	V.IAV	41	60.7	41.5	55.6	55.2	39.0	51.0
	V.LVC.full	55	34.6	16.4	28.3	45.8	20.0	**36.4**
	V.VID	47	55.6	21.3	42.0	62.5	21.3	**45.1**
	V.VPC.full	25	38.5	20.0	32.5	54.6	24.0	**43.5**
	V.VPC.semi	12	50.0	25.0	41.7	60.0	25.0	**46.9**
BART GEC	V.IAV	41	57.7	36.6	51.7	56.7	41.5	**52.8**
	V.LVC.full	55	43.3	23.6	37.1	42.9	21.8	35.9
	V.VID	47	55.6	21.3	42.0	78.6	23.4	**53.4**
	V.VPC.full	25	31.6	24.0	29.7	41.7	40.0	**41.3**
	V.VPC.semi	12	50.0	16.7	35.7	50.0	8.3	25.0

Table 5: GEC performance for different types of verbal MWEs.

terms of precision (P), recall (R) and $F_{0.5}$ using the ERRANT evaluation framework. $F_{0.5}$, which weights precision twice as much as recall, has been the most common evaluation metric for GEC since the CoNLL-2014 shared task (Ng et al., 2014).

Table 3 shows the overall performance of the encoder decoder GEC system (Experiment 1) in different settings: baseline (no MWE information), MWE-augmented [3-class] (auxiliary IOB MWE labels), MWE-augmented [23-class] (auxiliary IOB+type MWE labels). We can see that adding MWE information improves GEC system performance.

Table 4 shows the overall performance of the BART model (Experiment 2) fine-tuned on the standard W&I GEC data compared to the models trained on the MWE tagged data in scenarios 1 (where we predict MWEs in the original side) and 2 (where we predict MWEs in the corrected side). Overall, we see a slight improvement on the $F_{0.5}$ performance only in the case of MWE-augmented model (2).

4.5. Fine-grained analysis

We furthermore analyse the performance of our GEC models for specific types of MWEs. In particular, we aim to determine whether our systems are able to detect and correct incorrect usages of MWEs by learners. In order to perform this evaluation, we annotate our system output with MWE tags using the MWE identification system (Section 3) and find the overlap between MWE spans and ERRANT edit spans to determine which hypothesis edits involve MWEs. In this way, we can compare how our system performs on MWE errors irrespective of other errors. We particularly focus on verbal MWE errors.

Table 5 shows the results for both experiments. We focus on five types of verbal MWEs present in the corrected side of the data (V.IAV, V.LVC.full, V.VID, V.VPC.full, and V.VPC.semi) and compare the performance of the two GEC systems (encoder-decoder and BART) with or without MWE-augmentation. We focus on the 23-class MWE augmentation for the encoder-decoder system and scenario 2 for the BART system. In Table 5, we can see that GEC performance improves for four out of five verbal MWE types when we use MWE-augmented systems for the encoder-decoder GEC method and for three out of five verbal MWE types when we use setting 2 of the BART system. The highest improvement is in the case of VPC.full and VID, and the BART model results in more improvement (11.6 compared to 11 for VPC.full and 11.4 compared to 3.1 for VID). The BART system being unsuccessful in the case of V.LVC.full might be due to the fact that LVCs can have multiple arbitrary words in between their canonical form components (e.g. ***make a*** *very good* ***decision***). The marking system cannot differentiate them and considers words in between the MWE components as part of the MWE span which brings some noisy information to the system.

4.6. Discussion

In Table 6, we show two examples of sentences containing MWE errors corrected by each system. In the first example, none of the baseline systems were successful, but both MWE-augmented systems managed to correct the VPC *sign up*. In the second example, only the MWE-augmented BART system managed to correct the idiom *get to know*. This perhaps suggests the multi encoder-decoder system, which only uses MWE tags as token-level features, finds it hard to learn the notion of relationships between the components of the expression. The fact that we incorporate labels in the IOB labelling format combined with the MWE types helps the system have more informative features. However, the system still lacks direct linking information between MWE components. The BART system, on the other hand, has a different perspective and works with the text span representations that are encoded by special tokens. However it also treats all MWEs as continuous spans of texts of the same type and adds some arbitrary words in between their components. This is not favourable in the case of more structurally flexible MWEs such as LVCs. Non of the systems are yet successful in correcting more conceptual errors, for example in replacing *end up with* with *bring an end to* in the erroneous sentence, *'cars don't need necessarily to end up with the public transport'*.

	sentence
Original	*the course was fantastic and I am looking forward to signing it again next year .*
enc-dec	
baseline	the course was fantastic and I am looking forward to signing it again next year .
MWE-augmented	the course was fantastic and I am looking forward to **signing up** for it again next year .
BART	
baseline	the course was fantastic and I am looking forward to signing it again next year .
MWE-augmented	the course was fantastic and I am looking forward to **signing up** for it again next year .
Original	*it could allow you to communicate with people , know different cultures ...*
enc-dec	
baseline	it could allow you to communicate with people , know different cultures ...
MWE-augmented	it could allow you to communicate with people , know different cultures ...
BART	
baseline	it could allow you to communicate with people , know different cultures ...
MWE-augmented	it could allow you to communicate with people , **get to know** different cultures ...

Table 6: Example sentences with MWEs corrected by the encoder-decoder (enc-dec) and the BART MWE-augmented systems.

5. Conclusions

In this paper, we propose incorporating MWE information into two different GEC systems in order to improve GEC for MWEs which are challenging for language learners. The experiments show that the additions help GEC in the case of more conventional MWEs, like verbal idioms and verb particle constructions. More research is needed to improve GEC for more syntactically-flexible MWE types which allow arbitrary words in between their components. Our system relies on the performance of MWE detection systems as no GEC data is annotated for MWE type errors. This makes it more difficult for automatically correcting conceptual errors made by learners. Future work in this area benefits from more detailed annotation of learner errors related to their understanding of MWEs.

6. Bibliographical References

Awasthi, A., Sarawagi, S., Goyal, R., Ghosh, S., and Piratla, V. (2019). Parallel iterative edit models for local sequence transduction. 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)*, pages 4260–4270, Hong Kong, China, November. Association for Computational Linguistics.

Baldini Soares, L., FitzGerald, N., Ling, J., and Kwiatkowski, T. (2019). Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2895–2905, Florence, Italy, July. Association for Computational Linguistics.

Bryant, C., Felice, M., and Briscoe, T. (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, July. Association for Computational Linguistics.

Bryant, C., Felice, M., Andersen, Ø. E., and Briscoe, T. (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, August. Association for Computational Linguistics.

Christiansen, M. H. and Arnon, I. (2017). More than words: The role of multiword sequences in language learning and use. *Topics in Cognitive Science*, 9(3):542–551.

Clark, K., Luong, M.-T., Le, Q. V., and Manning, C. D. (2020). ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. In *International Conference on Learning Representations*.

Dahlmeier, D. and Ng, H. T. (2011). Correcting semantic collocation errors with L1-induced paraphrases. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 107–117, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Dahlmeier, D., Ng, H. T., and Wu, S. M. (2013). 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*, pages 22–31, Atlanta, Georgia, June. Association for Computational Linguistics.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Felice, M., Yuan, Z., Andersen, Ø. E., Yannakoudakis, H., and Kochmar, E. (2014). Grammatical error correction using hybrid systems and type filtering. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 15–24, Baltimore, Maryland, June. Association for Computational Linguistics.

Grundkiewicz, R., Junczys-Dowmunt, M., and Heafield, K. (2019). Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 252–263, Florence, Italy, August. Association for Computational Linguistics.

Katsumata, S. and Komachi, M. (2020). Stronger baselines for grammatical error correction using a pretrained encoder-decoder model. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 827–832, Suzhou, China, December. Association for Computational Linguistics.

Kochmar, E. (2016). Error detection in content word combinations. Technical report, University of Cambridge, Computer Laboratory.

Liu, N. F., Hershcovich, D., Kranzlein, M., and Schneider, N. (2021). Lexical semantic recognition. In *Proceedings of the 17th Workshop on Multiword Expressions (MWE 2021)*, pages 49–56, Online, August. Association for Computational Linguistics.

Meunier, F. and Granger, S. (2008). *Phraseology in foreign language learning and teaching*. John Benjamins Publishing.

Mizumoto, T., Mita, M., and Matsumoto, Y. (2015). Grammatical error correction considering multiword expressions. In *Proceedings of the 2nd Workshop on Natural Language Processing Techniques for Educational Applications*, pages 82–86, Beijing, China, July. Association for Computational Linguistics.

Nesselhauf, N. (2003). The Use of Collocations by Advanced Learners of English and Some Implications for Teaching. *Applied Linguistics*, 24(2):223–242, 06.

Ng, H. T., Wu, S. M., Briscoe, T., Hadiwinoto, C., Susanto, R. H., and Bryant, C. (2014). The conll-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14.

Nicholls, D. (2003). The cambridge learner corpus: Error coding and analysis for lexicography and ELT. In *Proceedings of the Corpus Linguistics 2003 conference*, pages 572–581.

Omelianchuk, K., Atrasevych, V., Chernodub, A., and Skurzhanskyi, O. (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, July. Association for Computational Linguistics.

Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., and Auli, M. (2019). fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Ramisch, C., Cordeiro, S. R., Savary, A., Vincze, V., Barbu Mititelu, V., Bhatia, A., Buljan, M., Candito, M., Gantar, P., Giouli, V., Güngör, T., Hawwari, A., Iñurrieta, U., Kovalevskaitė, J., Krek, S., Lichte, T., Liebeskind, C., Monti, J., Parra Escartín, C., QasemiZadeh, B., Ramisch, R., Schneider, N., Stoyanova, I., Vaidya, A., and Walsh, A. (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, August. Association for Computational Linguistics.

Sag, I. A., Baldwin, T., Bond, F., Copestake, A., and Flickinger, D. (2002). Multiword expressions: A pain in the neck for nlp. *Lecture Notes in Computer Science*, 2276:1–15.

Savary, A., Ramisch, C., Cordeiro, S., Sangati, F., Vincze, V., QasemiZadeh, B., Candito, M., Cap, F., Giouli, V., Stoyanova, I., and Doucet, A. (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, April. Association for Computational Linguistics.

Schneider, N., Onuffer, S., Kazour, N., Danchik, E., Mordowanec, M. T., Conrad, H., and Smith, N. A. (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, May. European Language Resources Association (ELRA).

Shwartz, V. and Dagan, I. (2019). Still a pain in the neck: Evaluating text representations on lexical composition. *Transactions of the Association for Computational Linguistics*, 7:403–419, March.

Siyanova, A. and Schmitt, N. (2007). Native and nonnative use of multi-word vs. one-word verbs. 45(2):119–139.

Stahlberg, F. and Kumar, S. (2020). Seq2Edits: Sequence transduction using span-level edit operations. In *Proceedings of the 2020 Conference on Empirical*

Methods in Natural Language Processing (EMNLP), pages 5147–5159, Online, November. Association for Computational Linguistics.

Stahlberg, F. and Kumar, S. (2021). Synthetic data generation for grammatical error correction with tagged corruption models. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 37–47, Online, April. Association for Computational Linguistics.

Taslimipoor, S., Bahaadini, S., and Kochmar, E. (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.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In I. Guyon, et al., editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Yannakoudakis, H., Briscoe, T., and Medlock, B. (2011). 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*, pages 180–189, Portland, Oregon, USA, June. Association for Computational Linguistics.

Yuan, Z. and Briscoe, T. (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, June. Association for Computational Linguistics.

Yuan, Z. and Bryant, C. (2021). Document-level grammatical error correction. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 75–84, Online, April. Association for Computational Linguistics.

Yuan, Z., Taslimipoor, S., Davis, C., and Bryant, C. (2021). Multi-class grammatical error detection for correction: A tale of two systems. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8722–8736, Online and Punta Cana, Dominican Republic, November. Association for Computational Linguistics.