Classifying Syntactic Errors in Learner Language

Leshem Choshen*
Department of Computer Science
Hebrew University of Jerusalem
leshem.choshen@mail.huji.ac.il

Dmitry Nikolaev*
Department of Linguistics
Stockholm University
dnikolaev@fastmail.com

Yevgeni Berzak
BCS
MIT
berzak@mit.edu

Omri Abend
Department of Computer Science
Hebrew University of Jerusalem
omri.abend@mail.huji.ac.il

Abstract

We present a method for classifying syntactic errors in learner language, namely errors whose correction alters the morphosyntactic structure of a sentence. The methodology builds on the established Universal Dependencies syntactic representation scheme, and provides complementary information to other error-classification systems. Unlike existing error classification methods, our method is applicable across languages, which we showcase by producing a detailed picture of syntactic errors in learner English and learner Russian. We further demonstrate the utility of the methodology for analyzing the outputs of leading Grammatical Error Correction (GEC) systems.

1 Introduction

Taxonomies of grammatical errors are important for linguistic and computational analysis of learner language, as well as for Grammatical Error Correction (GEC) systems.[1] Such taxonomies divide the complex space of errors into meaningful categories and enable characterizing their distribution in learner productions. This information can be beneficial for GEC: it can support the development of systems that focus on specific error types, serve as a form of inductive bias (for example, by regularizing the system's output to have a desired distribution over correction types), and guide data augmentation and data filtering by controlling the distribution of error types. Error taxonomies can also improve the interpretability of system outputs for error analysis and learner feedback.

A number of annotation efforts for learner language developed error taxonomies (Nicholls, 2003; Dahlmeier et al., 2013), and statistical classifiers into such taxonomies, notably ERRANT (Bryant et al., 2017). Taking error types into consideration in learning has also been shown to improve GEC performance (Kantor et al., 2019, cf. §6). However, most existing taxonomies are fairly coarse-grained and language specific, and do not produce meaningful types for a large proportion of the errors. For example, 25% of the errors in the standard NUCLE corpus (Dahlmeier et al., 2013) are mapped to the residual category OTHER (see §3.3).

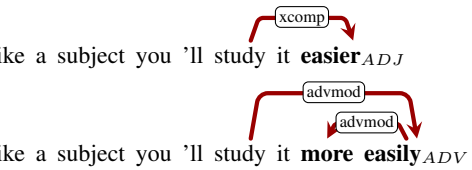

Figure 1: Example of an edit of type ADJ → ADV in POS terms and `xcomp` → `advmod` in edge-label terms. Corresponding spans are boldfaced.

We propose SERCL, a taxonomy of ***Syntactic Errors*** (SEs) and an automatic **Cl**assification. Inspired by a longstanding tradition in Machine Translation (MT) which analyses divergences between source and translated texts based on syntactic structure (Dorr, 1994; Nikolaev et al., 2020), SERCL is based on divergences between ungrammatical sentences and their corrections. We define SEs as errors whose correction involves changing morphological features, POS labels or the syntactic structure labels. SERCL takes as input edits, i.e., grammatically incorrect text spans and their corrections, and compares their labels. For example, the error in Fig. 1 is an adjective replaced with an adverb (ADJ→ADV) in POS terms, and an `xcomp`→`advmod` in edge-label terms. Thus, SEs are defined by changes in form, rather than by the principles governing the choice of a correct form.

SERCL is the first taxonomy derived from a syntactic representation framework, and it uses the Universal Dependencies formalism (UD; Nivre et al., 2016). This approach provides three ma-

*First two authors contributed equally.

[1]Code can be found in github repo GEC_UD_divergences. Matrices directly mentioned are included in the appendix.

Proceedings of the 24th Conference on Computational Natural Language Learning, pages 97–107

https://doi.org/10.18653/v1/P17

jor advantages over prior learner error taxonomies. First, the SERCL taxonomy is derived automatically from UD annotations, circumventing the need for constructing ad-hoc manually defined error categories. Second, using the UD formalism makes the method applicable across languages, allowing for consistent analyses and comparisons of learner errors across different languages within one unified framework. Third, SERCL is compatible with standard representations and tools in NLP.

Further, the UD based approach to error classification can yield finer distinctions compared to existing schemes. For example, it divides the commonly used class of adposition errors into errors in the use of prepositions as nominal modifiers (e.g., "a mention ~~to~~ **of** previous work"), and the use of prepositions in prepositional objects or adjuncts (e.g., "referring ~~for~~ **to** previous work"). POS tags alone cannot distinguish them, but the UD trees expose this distinction straightforwardly. UD can also help classify agreement and case-assignment errors thanks to its morphological-feature layer containing information about case, number, gender, and other features relevant for inflection.

We validate SERCL's reliability by showing (1) SEs based on automatic parses are similar to ones based on manual parses. (§3.1); (2) SERCL types map well to NUCLE's manually curated taxonomy (§3.2); (3) SERCL is complementary to the standard type classifier ERRANT: 60% of the errors not classified by ERRANT are classified by SERCL.

We demonstrate SERCL's unique features, notably cross-linguistic applicability, by analyzing SE distributions in available corpora for learner English (§4.1) and learner Russian (§4.2).

Finally, we find in GEC systems (1) certain SEs are harder to correct (2) SEs are harder than non-SEs (c.f. 5) (3) the granular types can help devising rules to improve products (e.g. Grammarly, §5.2).

2 Methodology

This section defines our taxonomy of SEs and how SERCL classifies into it. Given a parsed learner sentence and its correction, and given an edit $e = (e_s, e_c)$, i.e., a sub-string of the source sentence e_s that contains a grammatical error and its reference correction e_c, we define its class in the following way. We select a representative token in e_s and in e_c. Specifically, each sub-string defines a sub-forest of the dependency parse, and the representative is taken to be the node closest to the root.[2] The rationale for this decision is that UD treats grammatical markers as dependents of content words. Therefore, in most cases the semantic and syntactic heads correspond to one another, even if lexical items are changed. For example, in ***went***→*was* ***walking***, the semantic and syntactic head of the target has the lemma *walk* and not *be*.[3]

Acronym + Ref.	Notes
TLE (Berzak et al., 2016)	Manual parses
NUCLE (Dahlmeier et al., 2013)	Standard GEC benchmark
Lang8 (Mizumoto et al., 2012)	No error classes
W&I (Bryant et al., 2019)	Varied proficiency levels ERRANT classes
RULEC (Rozovskaya and Roth, 2019)	Learner Russian

Table 1: Datasets used in this work.

We define an SE as an edit where the two representative's labels do not match. The SE type is defined as the ordered pair of labels with the source label going first. Special cases of SEs are additions and deletions, i.e., edits in which the source or target span is empty.

This definition of SEs is straightforward to implement and requires no further annotation on top of the edits and parses, but it leaves out cases where the representative tokens retain their labels (e.g., agreement errors or inappropriate determiners), although these distinctions can be made in some cases based on UD's morphological features. For practical use, one can annotate all these non-SE errors by the feature that is retained (e.g. Plural Noun errors, if POS tag and morphological features are used). Given a corpus, a confusion matrix could be extracted, where the diagonal counts the non-SEs.

We focus in this work on universal POS-tag pairs, which are sufficient to classify and explain the majority of SEs in English. Dependency labels are analyzed as well, although we find that edge-label-based types and POS-based types are strongly correlated (§4.1). We also explore the use of morphological features, and apply it to Russian that has a rich morphology (§4.2).

[2] We select the leftmost token to break ties (3.5% of TLE SEs).

[3] In order to investigate the type correspondences of other tokens in the sub-strings, we may extract dependents of the representative nodes and compare their labels. This is an avenue for future work.

3 Reliability

3.1 Reliance on Automatic Parses

SERCL relies on syntactic trees. Manual annotation is currently only available for the TLE corpus (Berzak et al., 2016, all datasets addressed in the paper are summarized in Table 1), which includes POS and dependency relations, but no morphological features. Hence, we assess the outcomes of using a UD parser instead. We use UDPiPE (Straka et al., 2016) as our parser of choice. It is among the top-scoring parsers in CoNLL 2018 shared task (Zeman et al., 2018) for both English and Russian, the languages we consider in this paper.[4]

We begin by comparing the distribution of SEs in automatically parsed TLE and manual TLE. We focus the discussion on POS-based SEs for concisness; the full distributions of SE types, both edge-label-based and POS-based are found in Appendix §3.1. When normalising by the number of tokens per POS, class frequencies are quite close to those obtained by manual edits (0.4% absolute change on average and Pearson correlation of $r = 0.998$). This is also the case when normalising by the amount of SEs per POS (0.05% change). The results suggest that the use of a parser does not qualitatively change the distribution of SEs, and that current UD parsing technologies are mature enough to be used for extracting SEs.

While trends are similar with manual and automatic parses, perhaps unsurprisingly, more SEs are found when using automatic parses. This is particularly clear for the "other" tag "X" and for interjections. Symbols are the only category where we find less SEs. We ignore these non-lexical tags in our analysis, suspecting that this is a weakness of the parser. Finding the parsing reliable, we move to compare SERCL to existing approaches.

3.2 Comparing to Manually Typed Edits

Unlike many NLP tasks, this work does not aim to mimic human behavior. Still, there is sense in comparing SERCL to a manually annotated taxonomy. We compare NUCLE annotated train errors and SERCL's (confusion matrix in appendix Table 17). We ignore relocation errors as edits lack the necessary information to discern relocation from deletion.

[4] A number of works designed parsers with learner language specifically in mind. However, as such parsers exist only for learner English, we use UDPipe for uniformity.

Source POS	Target POS	#	Source label	Target label	#
NOUN	VERB	51	compound	amod	32
NOUN	ADJ	50	cop	aux	32
ADJ	NOUN	49	xcomp	ccomp	32
VERB	NOUN	46	obl	obj	26
VERB	ADJ	37	obl	advmod	25
DET	PRON	34	det	nmod:poss	25
PRON	DET	32	advmod	obl	24

Table 2: Most prevalent types of SEs involving replacement in the TLE in terms of POS tags (left) and edge labels (right). Numbers are absolute counts. See §2 in the supplementary material for example sentences.

SE types are generally contained within a single NUCLE error type. Indeed, on average 62% of the instances of a given SE type are contained in the maximally overlapping NUCLE category, i.e., when assigning each SE a NUCLE category most of the SE's instances are NUCLE's category instances as well. 82% of the instances on average belong to one of the three maximally overlapping NUCLE categories. CCONJ→ADV, for example, is almost solely (95%) mapped to "transition" error type, addressing linking and phrase errors. This shows that SE types contain much of the information conveyed by NUCLE types. Qualitatively, SERCL has more categories and splits NUCLE types to meaningful sub-types. It is thus usually more informative. For example, the "article or determiner" NUCLE type is split to insertions and deletions of determiners in addition to other SEs (mostly from or to determiner).

3.3 Comparing to the Automatic ERRANT

This section studies the relation between SERCL's predictions and those of ERRANT. For comparability, we apply SERCL to the edit spans produced by ERRANT. For brevity, we focus on POS-based SEs.

ERRANT (Bryant et al., 2017) is essentially the only classifier in use today, and is therefore a natural point of comparison. ERRANT taxonomy is coarse-grained. It assumes for the most part that POS tags are not altered in corrections, classifying many errors by their POS tag (e.g. adverb error). Consequently, ERRANT covers mostly spelling and word-form errors.

We note three important differences between SERCL and ERRANT. First, being based on UD, SERCL is applicable across languages (see §4.2), while ERRANT requires new rules or other modifications per language (Boyd, 2018). Second, relying

on an established framework with broad usability accords validity to SERCL's taxonomy, which is otherwise hard to validate (Bryant et al., 2017). Last, ERRANT classifies most SEs as OTHER. SERCL therefore complements ERRANT and is able to classify what ERRANT leaves unclassified.

Empirically, we find that ERRANT does not meaningfully classify a large portion of the errors: about 25% of ERRANT's predictions fall into the residual category Other in NUCLE and Lang8, and about 15% of them in W&I and TLE. We analyze which of those Other edits are SEs, finding most of them are. In W&I, of the 842 errors classified as OTHER, only 338 errors (40.1%) are cases where the POS remains unaltered, while the remaining 504 errors (59.9%) are POS-based SEs. The effective number of SE types that OTHER classifies into is 80.6, i.e., an entropy of 4.4 nets of the POS-based type distribution in edits classified as OTHER.

As for SEs not classified as OTHER, our manual analysis reveals that there too SERCL provides complementary information to ERRANT. Of the remaining errors, 620 are POS-based SEs, while 3211 are not (19.3%). Leaving out errors that involve punctuation leaves us with 522 SEs in W&I. Of those, the most common class is "morphological inflection" (indicating that the correction and the source share a lemma). On it, SERCL provides additional information, e.g., that the most frequent morphological inflection SE is NOUN→ADJ (31% of the cases), while the reverse direction is much rarer (7%). The second most common type, spelling, proved to be challenging for the parser and ERRANT is hence more informative for those. This is also the case for word-order errors. While verb errors are only the third most common, together with its subcategories, such as VERB:FORM, they account for 131 SEs. These might benefit from the SE categorization of common cases (e.g., VERB↔AUX errors suggest an error in the syntactic structure, unlike non-SE errors that usually involve lexical selection). Similarly, the 56 orthography errors could benefit from subcategorization of the common errors. For example, NOUN→PROPN is a common orthography error by ERRANT; ERRANT's type thus does not specify that it is a proper noun lacking capitalization. The other cases are either similar in spirit and can benefit from categorization of frequently appearing SEs, or cases where the POS tagging of the source and target disagree, either due to the UD guidelines or to parser inconsistency (e.g., the source parse may consider a word a particle, while the target parse considers it an adposition).

To conclude, about 60% of the errors classified as OTHER by ERRANT receive a POS-based SE class. SE classification further provides non-trivial information in many instances of other ERRANT categories. Together, these demonstrate that SERCL provides value beyond ERRANT's classification, even where only English is considered.

4 Cross-linguistic Corpus Studies

In this section, we apply SERCL to available datasets, comparing between different English datasets, originally annotated in different taxonomies, and between English and Russian datasets.

4.1 English

We analyze the English datasets and learner language characteristics through SEs. We start with TLE, which provides manual UD and edit annotation. Our analysis is based upon the available tokenization and edit annotations. To avoid double-counting, we merge overlapping edits to form a set of non-overlapping ones. After removing some noise in the XML markup, we extract 4584 SEs. Of those, 2042 are additions, 1048 are deletions, and 1495 are replacements. In 657 cases of replacement, both the POS and edge label are changed; in 306 cases, only the POS is changed; in 532 cases, only the edge-label is changed.

Figure 2 presents the most frequent addition and deletion types. Frequent POS tags are often frequently deleted or added POS tags, but not necessarily (e.g., nouns are almost twice as frequent as determiners). Additions are drastically more frequent than deletions for determiners, punctuation and pronouns and only slightly more for adpositions. Thus, we replicate the results that learners omit more than they add (Bryant et al., 2019), and give a detailed view on where they do not.

It is often straightforward to connect major types of POS and edge-label additions and deletions: the relationship between DET and `det` is trivial, and missing/redundant adpositions mostly correspond to `mark` and `case`. Deletions and additions of lexical categories with more variegated syntactic functions (such as nouns and verbs) correspond to more varied edge labels. Generally, however, changes in edge labels and POS tags are found to

	A	B	C	Native
SCONJ	0.804	0.864	0.923	0.942
DET	0.857	0.907	0.960	0.971
ADV	0.844	0.893	0.945	0.950
ADJ	0.875	0.923	0.962	0.972
ADP	0.891	0.935	0.969	0.976
PART	0.887	0.924	0.963	0.985
AUX	0.901	0.943	0.973	0.987
PROPN	0.902	0.930	0.966	0.968
NUM	0.897	0.929	0.960	0.950
PRON	0.908	0.930	0.963	0.953
NOUN	0.934	0.963	0.983	0.983
CCONJ	0.922	0.944	0.968	0.971
VERB	0.945	0.964	0.983	0.980
PUNCT	0.978	0.980	0.990	0.981

Table 3: Percentage of unchanged POS tags per type (rows) and proficiency level (columns) in the W&I dataset. Proficiency levels are A-C where C is the most proficient, the last column is for native speakers. Sorted by the average of columns A-C.

be highly correlated both for additions and deletions (Cramer's V = 0.78 for both categories) and replacements (Cramer's V = 0.76).

Most prevalent types of replacements are presented in Table 2. These may suggest a direction to focus GEC efforts towards, a direction we explore in §5.2. Full matrices are in appendix §3.1; incidentally, 44.4% of the errors are POS SEs.

Investigating SEs across levels (see Table 3), we find the most error-prone SEs are among the least difficult to natives. However, on the easiest SEs advanced learners outperform natives. Being out of scope, we leave the details, as well as comparison between levels across datasets to Appendix 1.

4.2 Russian

To demonstrate the generality of the proposed approach, we apply SERCL to RULEC, a corpus of learner Russian (Rozovskaya and Roth, 2019). Russian syntax is characterised by pervasive agreement and complex rules of case selection for nouns. UD morphological features, parsed by UDPipe, make it possible to analyze learners' errors arising due to these phenomena; they are taken up in §4.2.2.

4.2.1 POS mismatches in learners' Russian

An overview of POS additions and deletions is presented in Figure 3. Compared to English (cf. §4.1), learners of Russian tend to more actively underuse nouns (177 additions vs. 64 deletions) and pronouns (111 additions vs. 34 deletions). The latter may stem from Russian being a pro-drop language where subject pronouns can be omitted in certain contexts. The precise rules, however, are rather

	Acc	**Dat**	**Gen**	**Ins**	**Loc**	**Nom**
Acc	0	46	132	43	96	40
Dat	17	0	25	21	7	11
Gen	78	45	0	71	73	83
Ins	19	19	53	0	16	15
Loc	66	14	62	12	0	7
Nom	88	19	163	75	22	0

Table 4: Case corrections in nouns in Russian learners' sentences. Source(rows) against reference (columns).

complicated, and it takes a lot of practice knowing when the result sounds felicitous (Zdorenko, 2010).

Most dominant types of POS replacements are similar (ADJ→NOUN, 80 cases; NOUN→ADJ, 75; VERB→NOUN, 65, PRON→DET, 50; NOUN→VERB, 45); however, ADV→ADJ (66) and ADJ→ADV (51) are also prominent, which may be since adjectives and adverbs are more strictly distinct in Russian.

4.2.2 Morphological Features

Russian possess a mildly complex conjugation and inflection system, which leaves a lot of room for errors even in cases when a correct POS is selected. The feature layer of UD makes it possible to identify these errors, which are dominated by three large classes: agreement errors (wrong person/number/gender features on verbs and wrong number/gender/case features on adjectives), case-assignment errors on nouns and pronouns, and verbal errors regarding aspect and voice.

A breakdown of case-assignment errors for nouns is presented in Table 4. It shows, among other things, that learners tend to use accusative and nominative cases in contexts where Russian demands the genitive case (which, in addition to the cross-linguistically frequent possessive meaning, also has numerous more subtle uses, e.g. in some types of negative sentences). Case-agreement errors on adjectives, on the other hand, tend to be more symmetric: 27 cases of an accusative case ending instead of a genitive vs. 19 cases of the converse error (see confusion matrices for case, gender, and number on adjectives in Appendix §3.5).

The Russian verbal system also presents learners with several difficulties that our analysis echoes. Verbs fall into two aspectual classes (with *perfective* verbs denoting completed actions and *imperfective* verbs actions-in-progress and habitual actions), and it is difficult for learners with native languages lacking this distinction to use them correctly (e.g., the English phrase *I went to work* will be translated differently depending on whether it is modified by *yesterday* or *every day*). The feature analysis shows

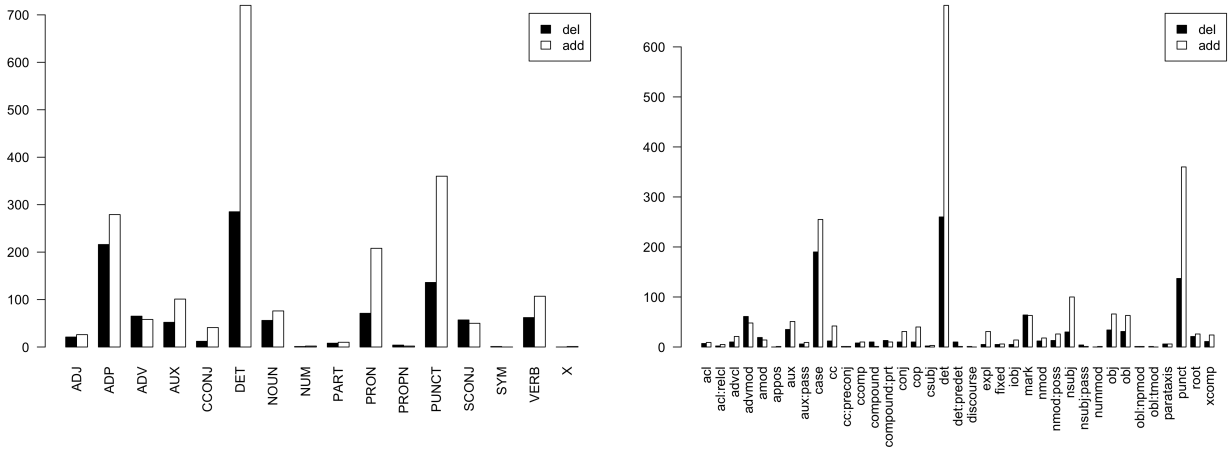

Figure 2: Left: POS tags of words deleted or added in corrected sentences in absolute counts; Right: edge labels of words deleted or added in corrected sentences in absolute counts (y).

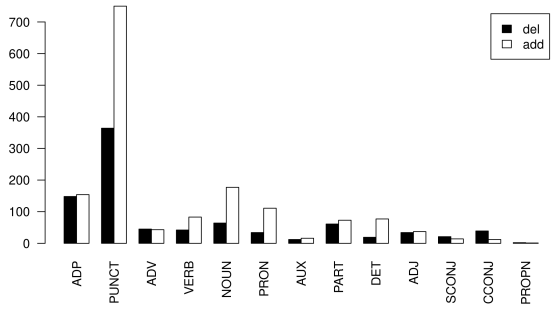

Figure 3: POS tags of words deleted or added in corrected Russian sentences.

that incorrect perfective verbs were changed into imperfective ones 210 times, and imperfective was changed into perfective 223 times.

Another complication stems from the use of middle voice in Russian. English is dominated by *labile* verbs denoting both spontaneous and caused actions (*The cup broke* vs. *I broke the cup*). In Russian, verbs for spontaneous actions are usually derived from transitive verbs (*razbil*[broke.smth]→*razbil**ja***[got.broken]). Derived intransitive verbs are analyzed as "middle voice", and the inspection of verbal voice mismatches shows that the dominant type of error is the use of active voice instead of middle voice (108 cases vs. 45 cases of the converse error). These findings can inform future efforts of GEC evaluation and development, in addressing these recurring error patterns.

5 Analyzing GEC System Outputs

To further showcase the utility of SERCL, we demonstrate GEC system analysis with it.

	AIP-TOHOKU	UEDIN-MS	GOLD
ADJ→ADV	18	21	55
ADJ→NOUN	37	46	105
ADJ→VERB	21	33	59
NOUN →VERB	76	87	142
VERB→ADJ	17	21	38
NUM→DET	3	2	5
PART→DET	0	3	1

Table 5: AIP-TOHOKU, UEDIN-MS and Gold annotation changes (correct or not) on selected replacement types of SEs in absolute counts. The non-uniform behaviour of the systems over types indicates that SERCL produces meaningful results.

5.1 Experiments with Leading GEC Systems

We use the outputs of several systems that participated in the BEA2019 shared task (Bryant et al., 2019), namely: the winning system UEDIN-MS (Grundkiewicz et al., 2019), as well as KAKAO&BRAIN (Choe et al., 2019), SHUYAO (Xu et al., 2019), CAMB-CUED (Stahlberg and Byrne, 2019), and AIP-TOHOKU (Asano et al., 2019), that were ranked second, fifth, eighth and ninth respectively.[5] We extract matrices for the system outputs using the same method as in §3.1. Recall is bounded by the amount of predicted SEs, divided by their number in the gold standard. The full matrices are given in Appendix §3.2.

Our results in Table 5 show that the top-ranking UEDIN-MS makes consistently more changes in general and per source SE than AIP-TOHOKU ranked 9th, but less than CAMB-CUED, ranked 8th (found in appendix §3.2). However, there is no correlation between the number of SE changes in general or per source tag and the rank of the system

[5]The outputs will be published with the rest of our code, as they were deleted from the contests' page. We thank Yoav Kantor for providing us with the data.

(both yield a partial-order Kendall $\tau = 0$). This is in line with Choshen and Abend's (2018b), who found no relation between a system's performance and its conservatism.

Some SE types are harder for the examined systems than others. There is a slight negative correlation ($r = -0.16$) between the average recall bound of the systems and frequency in the gold standard. For example, pronouns are well addressed across systems, with 65% average recall bound, while numerals are less so, with 43%. This implies that less frequent corrections are handled less well, but also opens room for improvement. An example is numerals and coordinating conjunctions, which are handled less well.

	UEDIN-MS	GOLD	Ratio (%)
CCONJ	71	158	44.9
NUM	22	48	45.8
SCONJ	114	233	48.9
AUX	153	310	49.4
VERB	202	405	49.9
ADP	232	461	50.3
PROPN	62	114	54.4
NOUN	346	618	56.0
ADV	273	472	57.8
PART	97	166	58.4
PUNCT	116	191	60.7
ADJ	283	462	61.3
DET	348	568	61.3
PRON	367	584	62.8
Overall	2686	4790	56.1

Table 6: Amount of syntactic changes per source POS tags for UEDIN-MS and the gold standard in absolute counts. The ratio is an upper bound on the recall.

We further revisit the replacement types discussed in §4.1 and compute the recall upper bound for the top system UEDIN-MS (Table 5 and aggregation per source POS in 6). Putting aside the rare types NUM→DET and PART→DET, we find that UEDIN-MS tackles considerably more SEs than Grammarly (see §5.2 below and Table 7). However, the recall bound is not uniform across types, where two types receive very low recall (ADJ→ADV with 38% and ADJ→NOUN with 44%), indicating these as potential directions for future work.

The bound over all SEs for UEDIN-MS is 56% (50% on the subset of errors discussed above). If we assume the precision on SE is similar to the overall reported precision (72%), we may conclude that recall for SEs is around 40%. For comparison, the overall reported recall is 60%, which suggests that SEs are harder on average than non-SEs, underscoring the value in classifying them.

5.2 Prospects for Improving GEC using Fine-grained SE Classification

To demonstrate the benefits of fine-grained SE categories, we analyze several SE types involving word replacements that are not as prevalent in TLE as additions and deletions of determiners and prepositions, but are still recurrent and form a closed-class that is likely to be addressable through designated GEC modules. We also consider the open-class VERB→ADJ for comparison. We examine the performance of a leading GEC tool, Grammarly in handling these types and analyze the capabilities of end-to-end systems in §5.

NUM→DET. Almost all examples include *One* instead of *any* or *another*. Example: *Technology is also important in ~~one~~ another area for me.*

PART→DET. All examples show *not* used instead of *no*. Example: *There was ~~not~~ no discount*.

ADJ→NOUN. Such replacements mostly involve quantifiers (*many →a lot of*, *small →a few*) which constitute a closed class. Example: *But some schools in my country don't allow ~~many~~ a lot of things.* Another subtype of this SE is more open-ended in that it involves using adjectives instead of morphologically related nouns (e.g., *joyful→joy*, *late →lateness*). Example: *Second, there should be friends and family members in the home to provide ~~joyful~~ joy and fun.* These SEs should be easy to detect because derivational relations are mostly transparent. However, there are a handful of harder cases (e.g., *I felt like a ~~dumb~~ fool*) whose correction demands more nuanced lexical knowledge.

NOUN→VERB. This SE type usually involves a morphologically-related form (*entrance→enter*, *product→produce*). However, some of the examples of this type are ambiguous due to English zero-derivation of deverbal nouns. Cf. *I love sleep in tents*, where *to sleep* and *sleeping* are both valid corrections found in the corpus. Example: *When we ~~entrance~~ entered the place our problems began.*

VERB→ADJ. Those replacements are diverse and often include large changes, mostly when the original sentence uses a completely wrong form of expression. Errors involving a verb rather a passive participle, which acts as an adjective, are also frequent (*trust was broken→betrayed trust*, *have conscience→are aware*, *problems involve with →problems involved with*).

		Amount	Detected	Valid	Precision	Recall
Add	None→ADV	58	0	0	0%	0%
	None→DET	44	13	10	77%	23%
	None→PRON	120	4	1	25%	1%
	None→VERB	107	0	0	0%	0%
Delete	ADV→None	64	3	0	0%	0%
	DET→None	49	20	20	100%	41%
	Pron→None	71	4	4	100%	6%
	VERB→None	41	4	0	0%	0%
Replace	ADJ→ADV	101	19	12	63%	12%
	ADJ→NOUN	45	4	2	50%	4%
	ADJ→VERB	18	0	0	0%	0%
	NOUN→VERB	44	10	8	80%	18%
	VERB→ADJ	26	3	0	0%	0%
	NUM→DET	7	0	0	0%	0%
	PART→DET	6	4	4	100%	67%

Table 7: Grammarly's performance on selected SE types in absolute counts. The varying behaviour per type indicates the separation to types is meaningful.

We turn to analyzing Grammarly's performance on the types discussed, as well as the four most frequent SE types of deletions, additions, and replacements (two of which are not among the above types). Grammarly's performance is of particular interest due to its reliance on designated modules (classifiers and rules) for addressing specific error types. Our results thus demonstrate how such a system may benefit from uncovering error types that can be addressed by integrating additional modules.

We manually annotate whether the edit in question is at all detected and whether it is validly corrected by Grammarly. As Grammarly may offer more than a single correction, we deem correct any case where at least one of the corrections is valid.

Results (Table 7) indicate that Grammarly fares poorly in addressing SEs, with the possible exception of superfluous determiners. Indeed, in many of the cases, only a small portion of the SEs was detected. While it is possible that Grammarly tends to overlook such cases because of the dominance of punctuation, spelling, and determiner errors in learner language, some of the types here involve only a handful of lexemes, suggesting that targeted treatment or data augmentation may be effective.

6 Related Work

Error types are often used to improve performance and evaluation in GEC. Taxonomies have been used to construct classifiers and rule-based engines to correct specific error types (e.g., Rozovskaya et al., 2014; Farra et al., 2014; Zheng et al., 2018). When using end-to-end systems, balancing the distribution of errors in the train and test sets has been shown to improve results (Junczys-Dowmunt et al., 2018). Ensembling black-box systems relying on per-type performance has been shown superior to each system's performance and over average ensembling (Kantor et al., 2019). Augmenting the training data with synthetic errors of a particular type is effective for improving performance on that type (Belinkov and Bisk, 2018). The classification of grammatical error types is also used to analyze system performance (e.g., Lichtarge et al., 2019). Choshen and Abend (2018a,b) showed that current systems and evaluation measures essentially ignore some error types, suggesting that targeted evaluation of these types may be needed.

To date, several error taxonomies have been proposed and applied for annotating errors in major English learner-language corpora (Bryant et al., 2019; Dahlmeier et al., 2013; Nicholls, 2003, *inter alia)*. There has been interest lately in other languages, for which different datasets and taxonomies were created (Rozovskaya and Roth, 2019; Rao et al., 2018; Zaghouani et al., 2014). However, different taxonomies are used by different corpora, based on commonly observed error types in the target domain and language, which impedes direct comparison across corpora. Moreover, these taxonomies are not formulated based on a specific theory or annotation scheme for morphosyntactic representation, which may promote accessibility to non-experts but often leads to non-uniform terminology and difficulty in leveraging available NLP tools.

Another automatic type classification was suggested apart from ERRANT. Swanson and Yamangil (2012) trained a log-linear model to predict types defined by Nicholls (2003). This taxonomy resembles ours in that it uses grammatical categories (POS tags), but differs in that it only distinguishes types based on the POS tag of the

correction and not of the source sentence. Moreover, relying solely on POS tags yields difficulties in classifying constructions that involve more than a single word. For such cases, it defines specialized error types, such as *Incorrect Argument Structure*, which serves as a residual category for argument structure errors that cannot be accounted for by adposition or agreement errors. However, unlike SERCL, it does not provide any information as to what particular incorrect argument structure was used or how it should be corrected.

Choshen and Abend (2018c) used a semantic annotation(Abend and Rappoport, 2013) to show semantics, unlike syntax is kept upon changes. UD was previously used in GEC in the TLE corpus and in a learner language parser (e.g., Sakaguchi et al., 2017) (we do not apply their parser, as it is made specifically for English, and might alter the origin parse).

7 Conclusion

We presented SERCL, a novel method for classifying SEs based on UD parses of learner text and its correction. We show that SERCL provides a detailed picture of the prevalence of different SEs in two languages, and can be straightforwardly automated. We further show that the method manages to classify about 60% of the unclassified edits by ERRANT, the standard tool for error classification, and provides useful complementary information for many of the classified edits.

Future work will combine SERCL and ERRANT into a single tool for English error classification (work in this direction has already begun). The experiments we presented show that several leading GEC systems of different types make errors of types that are not well-addressed by current systems. These results can inform the future development of tailored solutions for these cases.

Acknowledgments

We thank Yarden Gavish for her help with coding and data preparation assignments. This work was supported by the Israel Science Foundation (grant no. 929/17). Leshem Choshen is supported

References

Omri Abend and A. Rappoport. 2013. Universal conceptual cognitive annotation (UCCA). In *ACL*.

Hiroki Asano, Masato Mita, Tomoya Mizumoto, and Jun Suzuki. 2019. The AIP-tohoku system at the BEA-2019 shared task. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 176–182, Florence, Italy. Association for Computational Linguistics.

Yonatan Belinkov and Yonatan Bisk. 2018. Synthetic and natural noise both break neural machine translation. *ICLR*.

Yevgeni Berzak, Jessica Kenney, Carolyn Spadine, Jing Xian Wang, Lucia Lam, Keiko Sophie Mori, Sebastian Garza, and Boris Katz. 2016. Universal Dependencies for learner english. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 737–746.

Adriane Boyd. 2018. 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*, pages 79–84, Brussels, Belgium. Association for Computational Linguistics.

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 *ACL*.

Yo Joong Choe, Jiyeon Ham, Kyubyong Park, and Yeoil Yoon. 2019. A neural grammatical error correction system built on better pre-training and sequential transfer learning. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 213–227, Florence, Italy. Association for Computational Linguistics.

Leshem Choshen and Omri Abend. 2018a. Automatic metric validation for grammatical error correction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Leshem Choshen and Omri Abend. 2018b. Inherent biases in reference-based evaluation for grammatical error correction and text simplification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Leshem Choshen and Omri Abend. 2018c. Reference-less measure of faithfulness for grammatical error correction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Daniel Dahlmeier, Hwee Tou Ng, and Siew Mei Wu. 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.

Bonnie J. Dorr. 1994. Machine translation divergences: a formal description and proposed solution. *Computational linguistics*, 20(4):597–635.

Noura Farra, Nadi Tomeh, Alla Rozovskaya, and Nizar Habash. 2014. Generalized character-level spelling error correction. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 161–167.

Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Kenneth Heafield. 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. Association for Computational Linguistics.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Shubha Guha, and Kenneth Heafield. 2018. Approaching neural grammatical error correction as a low-resource machine translation task. In *NAACL-HLT*.

Yoav Kantor, Yoav Katz, Leshem Choshen, Naftali Naftali, and Noam Slonim. 2019. Learning to combine grammatical error corrections. In *BEA 2019 Shared Task: Grammatical Error Correction*.

Jared Lichtarge, Christopher Alberti, Shankar Kumar, Noam Shazeer, Niki Parmar, and Simon Tong. 2019. Corpora generation for grammatical error correction. In *NAACL-HLT*.

Tomoya Mizumoto, Yuta Hayashibe, Mamoru Komachi, Masaaki Nagata, and Yuji Matsumoto. 2012. The effect of learner corpus size in grammatical error correction of esl writings. In *COLING*.

Diane Nicholls. 2003. The cambridge learner corpus: Error coding and analysis for lexicography and elt. In *Proceedings of the Corpus Linguistics 2003 conference*, volume 16, pages 572–581.

Dmitry Nikolaev, Ofir Arviv, Taelin Karidi, Neta Kenneth, Veronika Mitnik, Lilja Maria Saeboe, and Omri Abend. 2020. Fine-grained analysis of cross-linguistic syntactic divergences. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1159–1176, Online. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proc. of LREC*.

Gaoqi Rao, Qi Gong, Baolin Zhang, and Endong Xun. 2018. Overview of nlptea-2018 share task chinese grammatical error diagnosis. In *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications*, pages 42–51.

Alla Rozovskaya and Dan Roth. 2019. Grammar error correction in morphologically rich languages: The case of russian. *Transactions of the Association for Computational Linguistics*, 7:1–17.

Alla Rozovskaya, Dan Roth, and Vivek Srikumar. 2014. Correcting grammatical verb errors. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 358–367.

Keisuke Sakaguchi, Matt Post, and Benjamin Van Durme. 2017. Error-repair dependency parsing for ungrammatical texts. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 189–195.

Felix Stahlberg and Bill Byrne. 2019. The CUED's grammatical error correction systems for BEA-2019. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 168–175, Florence, Italy. Association for Computational Linguistics.

Milan Straka, Jan Hajic, and Jana Straková. 2016. Udpipe: Trainable pipeline for processing conll-u files performing tokenization, morphological analysis, pos tagging and parsing. In *LREC*.

Ben Swanson and Elif Yamangil. 2012. Correction detection and error type selection as an ESL educational aid. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 357–361, Montréal, Canada. Association for Computational Linguistics.

Shuyao Xu, Jiehao Zhang, Jin Chen, and Long Qin. 2019. Erroneous data generation for grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 149–158, Florence, Italy. Association for Computational Linguistics.

Wajdi Zaghouani, Behrang Mohit, Nizar Habash, Ossama Obeid, Nadi Tomeh, Alla Rozovskaya, Noura Farra, Sarah Alkuhlani, and Kemal Oflazer. 2014. Large scale Arabic error annotation: Guidelines and framework. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland. European Language Resources Association (ELRA).

Tatiana Zdorenko. 2010. Subject omission in russian: a study of the russian national corpus. In Stefan Th. Gries, Stefanie Wulff, and Mark Davies, editors, *Corpus-linguistic applications. Current studies, new directions*, pages 119–133. Brill Rodopi.

Daniel Zeman, Jan Hajič, Martin Popel, Martin Potthast, Milan Straka, Filip Ginter, Joakim Nivre, and Slav Petrov. 2018. Conll 2018 shared task: multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–21.

Junchao Zheng, Courtney Napoles, Joel Tetreault, and Kostiantyn Omelianchuk. 2018. How do you correct run-on sentences it's not as easy as it seems. In *Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text*, pages 33–38.