Towards Automatic Grammatical Error Type Classification for Turkish

Harun Uz and **Gülşen Eryiğit**
İTÜ NLP Research Group
Faculty of Computer & Informatics
Istanbul Technical University
İstanbul, Türkiye
[uz20, gulsen.cebiroglu]@itu.edu.tr

Abstract

Automatic error type classification is an important process in both learner corpora creation and evaluation of large-scale grammatical error correction systems. Rule-based classifier approaches such as ERRANT have been widely used to classify edits between correct-erroneous sentence pairs into predefined error categories. However, the used error categories are far from being universal yielding many language specific variants of ERRANT. In this paper, we discuss the applicability of the previously introduced grammatical error types to an agglutinative language, Turkish. We suggest changes on current error categories and discuss a hierarchical structure to better suit the inflectional and derivational properties of this morphologically highly rich language. We also introduce ERRANT-TR, the first automatic error type classification toolkit for Turkish. ERRANT-TR currently uses a rule-based error type classification pipeline which relies on word level morphological information. Due to unavailability of learner corpora in Turkish, the proposed system is evaluated on a small set of 106 annotated sentences and its performance is measured as 77.04% $F_{0.5}$ score. The next step is to use ERRANT-TR for the development of a Turkish learner corpus. The code will be made publicly available.[1]

1 Introduction

Automatic error type classification is the task of assigning error type classes to predetermined grammatical errors. The Building Educational Applications (BEA) 2019 Shared Task on Grammatical Error Correction (GEC) (Bryant et al., 2019) emphasizes the importance of error correcting systems for educational applications. A total of 24 teams have developed GEC systems for the task and ERRANT (Bryant et al., 2017) (a rule-based error type classifier) was used for automatic evaluation of these systems on 5 different datasets. Automatic evaluation in GEC is the process of error classification on parallel data consisting of erroneous and correct sentence pairs. Automatic error classification is an important process while evaluating GEC systems, since the direct approach of exact match precision and recall scores is not intuitive enough to correctly analyze the strengths and weaknesses of these systems.

Due to the advances in deep learning based language processing in recent years, considerable performance improvements have been observed in GEC systems. In parallel with this, the need for bigger and labeled datasets increased even more. Usually, learner corpora[2] are used to create GEC datasets since foreign-language learners are the ones who make such grammatical errors the most. For resource-rich languages like English, there exist many such corpora; e.g., Cambridge Learner Corpus (Nicholls, 2003), NUCLE (Dahlmeier et al., 2013) and W&I+LOCNESS (Bryant et al., 2019). Collecting and annotating learner corpora are pretty costly and time-consuming tasks. The erroneous sentences need to be corrected by professionals, and the inter-annotator agreement should be high (annotators should use a universal set of error categories as much as possible) so that a sufficient amount of useful samples could be obtained. The challenges of creating a learner corpus have also led researchers to find alternative data resources such as extracting edit history of comments from the web (Chen et al., 2019). ERRANT-like systems help professionals annotate a vast amount of data quickly in a semi-automatic way.

Although there exist no prior complete GEC tools nor datasets for Turkish, there exist some related works (e.g., datasets related to social media errors of native speakers (Eryiğit and Torunoğlu-

[1] https://github.com/harunuz/erranttr.git

[2] Learner corpora are electronic collections of language data produced by L2 learners (second or foreign-language learners).

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop, pages 134–142

Selamet, 2017) or to some specific error type (Arikan et al., 2019)). These datasets are not directly usable in a general GEC system since they mostly focus on social media specific error types such as intentional repetition of the same characters for exclamation purposes or misuse of diacritics due to wrong keyboard choices.

Text normalization and spelling correction methods which have been developed with these datasets are evaluated according to the exact match scores. However, due to the rich morphological properties of Turkish, the exact match does not provide much insight on the type of errors where the system produces good and bad results. In Turkish, most of grammatical errors occur in suffixes as the learners tend to make inflectional and derivational errors Başak Karakoç Öztürk (2017). And in theory, since Turkish is a highly agglutinative language, it is possible to apply some derivations recursively and result in an infinite number of possible word derivations in Turkish. A single word in Turkish contains more syntactic information compared to English and corresponds to several English words most of the time. Therefore, while evaluating a Turkish GEC system output, a simple surface form matching approach loses more valuable information than it would in English.

In this paper, we discuss the applicability of the previously introduced ERRANT's grammatical error types to an agglutinative language, Turkish. We suggest changes on current error categories and discuss a hierarchical structure to better suit the inflectional and derivational properties of this morphologically highly rich language. The paper introduces ERRANT-TR, the first automatic error type classification toolkit for Turkish. ERRANT-TR currently uses a rule-based error type classification pipeline which relies on word level morphological information. Due to unavailability of learner corpora in Turkish, the proposed system is evaluated on a small set of 106 annotated sentences and its performance is measured as 77.04% $F_{0.5}$ score.

2 Related Work

Automatic error annotation has been a popular topic in computational linguistics for a long time (Wang et al., 2020; Bryant et al., 2022). Researchers tried to come up with standard error categories to cover possible error types in mono or multilingual settings and tried to develop automatic annotator systems. However, it is a challenging problem to come up with a unified solution and error categories due to the big structural differences between languages. There have been many attempts to develop ERRANT-like algorithms for different languages such as Greek (Korre et al., 2021), Czech (Náplava et al., 2022), German (Boyd, 2018), Spanish (Davidson et al., 2020), Russian (Katinskaia et al., 2022) and Korean (Yoon et al., 2022). Each work considers the original error types (Bryant et al., 2017) for the corresponding language and update them according to new needs. The need for developing particular annotation methods, even for languages that fall under the same language families, proves the ongoing challenge of developing a universal error annotation scheme.

Learner corpora have been the main focus when it comes to creating a GEC learning dataset recently. Synthetically generated data has been used prior to learner corpora and are still being used as additional data during the development of GEC systems (Kaneko et al., 2020; Kiyono et al., 2019; Zhao et al., 2019; Rothe et al., 2021; Omelianchuk et al., 2020; Lichtarge et al., 2019; Grundkiewicz et al., 2019). There has been other semi-automatic alternatives for creating learning datasets such as extracting Wikipedia edits (Grundkiewicz and Junczys-Dowmunt, 2014) as correct-erroneous sentence pairs. Both synthetically generated and semi-automatic datasets can be found in large amounts. However, they do not contain the natural distribution of real world errors. Therefore, learner corpora have become the de-facto data source for GEC systems, and many works focused on collecting and annotating these corpora (Katinskaia et al., 2022; Davidson et al., 2020; Boyd, 2018; Náplava et al., 2022). English has been the main subject of learner corpora studies as the language with the highest number of foreign students from all over the world. However for many other languages, these resources are still missing.

For Turkish, by the time of this writing, there is an ongoing research on collecting and manually annotating the first Turkish learner corpus. In the future, we plan on evaluating our system on this corpus when the data is publicly available.

3 ERRANT-TR

In this section, we present the first version of ERRANT-TR which relies on original ERRANT error categories described in Bryant et al. (2017). We explain how we discover these error types from the morphological structure of Turkish. There are 25 error types introduced with ERRANT. The list of categories and possible examples can be seen in Table 1.

In order to categorize the error types, we make use of the morphological information obtained through an automatic morphological analyzer and disambiguator (Akın and Akın, 2007). After the disambiguation process, we extract POS tags for each word from its morphological properties. The edits between correct and erroneous sentences can be grouped under 3 main categories at token level and the error types can be prefixed with "R", "M" and "U" labels, indicating "Replacement", "Missing" and "Unnecessary" errors respectively: A necessary token may be **M**issing or an **U**nnecessary token may be added in the erroneous sentence. But the most common, correct token(s) are **R**eplaced with erroneous token(s). Thus, not all error types are paired with "M" and "U". The possible combinations of labels can be seen in Bryant et al. (2017)'s Table 9. There can also be one-to-many or many-to-one alignments.

3.1 Edit Extraction

The first step to classify error types is to align the erroneous parts of the corrupt sentence with the correct parts of the correct sentence. ERRANT uses an edit extraction method introduced by Felice et al. (2016). It uses a modified Damerau-Levenshtein algorithm enhanced with linguistic features (POS tags, lemmas etc.) to extract the potential edits and merges some of them with predefined rules. In our experiments, with Turkish POS tag information added, it produced good results, therefore we used it as is.

3.2 Error Type Classification

The error type classification for each edit is done with a set of rules. The main information source for an edit classification is the morphological analysis and the POS tag of tokens in the correct sentence. An edit that does not contain an original token (an unnecessary token is used in the erroneous sentence) is difficult to identify as the only clue for the error type is the corrupt token(s). An edit that does not contain a corrupt token indicates a missing token error and the output purely relies on the success of the morphological analysis of the correct token(s). An edit that contains both correct and corrupt tokens is the most common alignment type and can have the most diverse set of error types.

Since in agglutinative languages most of the syntactic information resides at morphology level, the inflections are very rich and alignment at word level is not enough to specify the error types: one needs to align the morphemes (between correct and corrupt tokens) in order to specifically determine the error category (Yoon et al., 2022). As stated in Section-3.1, we use the default ERRANT aligner for word-level alignment. However, we use morphological features annotated at morpheme level in order to first align the morphemes and then specify the error types during error classification. Morphemes are aligned only if they are similar types (tense, mood, person, number etc.) or the similarity score for their surface forms exceeds a predefined threshold which is set to 0.85 by default. A sample output from the used morphological analyzer is provided below. The first line provides the correct and erroneous words respectively. The second line provides the morphological analysis of the correct word. The third line provides the aligned morphemes of the erroneous word. The word lemma is "*git*" but a probable stem "*gid*" is also provided by the used tool. Upon morpheme level alignment, we can observe that the tense suffix of the verb is produced erroneously:

```
gidiyorum   -> gidiyirum
git/gid(Verb)   iyor(Pres)  um(A1sg)
gid             iyir        um
 +               X          +
```

We used 50 erroneous sentences from Kurt (2020), Şahin (2013) and Fidan (2019) during the development of the classifier to validate the rules. The sentences were labeled by a linguist according to the error types in Table-1. The system tries to classify edits with the rules described in this section and assigns the discussed error types.

WO, ORTH and PUNCT errors (Table 1) are independent from morphological analysis and can be checked before the main decision mechanism. In order to classify these, we use ERRANT's methods as they are. Contractions (CONTR) in English combine a pronoun/noun and a verb, or a verb and the word "not", in a shorter form. CONTR errors are not common in Turkish. The words *"daha" (more)* and *"en" (the most)* are used before an adjective for comparative and superlative respectively, but differing from English, the adjective form itself is not affected with these constructions. Therefore, ADJ:FORM errors are also not common in Turkish.

Error Code	Meaning	Example
ADJ	Wrong choice of adjective	*büyük -> küçük*
ADJ:FORM	Wrong usage of comparative or superlative adjective	-
ADV	Wrong choice of adverb	*önce -> sonra*
CONJ	Wrong choice of conjunction	*ama -> belki*
CONTR	Wrong choice of contraction	-
DET	Wrong choice of determiner	*bu* elma -> *o* elma
MORPH	Tokens have the same lemma but nothing else in common	-
NOUN	Wrong choice of nouns	*kalem -> silgi*
NOUN:INFL	Count-mass noun errors	-
NOUN:NUM	Wrong usage of noun number	*elma -> elmalar*
NOUN:POSS	Wrong usage of noun possessive	*hastalarının ilaçları -> hastaların ilaçları*
ORTH	Case and/or whitespace errors	*herşey -> her şey*
OTHER	Errors that do not fall into any other category	-
PART	Wrong choice of particle	-
PREP	Wrong choice of preposition	*gibi -> için*
PRON	Wrong usage of pronoun	*sen -> ben*
PUNCT	Wrong usage of punctuation	*? -> !*
SPELL	Misspelling	*broblem -> problem*
UNK	A detected but not corrected error	-
VERB	Wrong choice of verbs	*geldim -> gittim*
VERB:FORM	Infinitives, gerunds and participles	gitmek, gitme, giden
VERB:INFL	Wrong usage of tense morphology	*(biz) yaptız -> (biz) yaptık*
VERB:SVA	Subject-verb agreement	*sen geliyorum -> sen geliyorsun*
VERB:TENSE	Wrong choice of inflectional and periphrastic tense, modal verbs and passivization	*geliyorum -> gelmiştim*
WO	Word order	*elma kırmızı -> kırmızı elma*

Table 1: Error code, description and examples. A dash indicates that the category has no example for being either too wide or not useful for Turkish. The original table is introduced in Bryant et al. (2017).

Both particles and prepositions (and postpositions as well) are considered as "edat" in Turkish. "Edat"s have a much broader scope and they may appear as either standalone words (e.g. *ile (with), için (for)*), or as suffixes (e.g. *-le (with)*) or as both suffix and a word (e.g. *-a kadar (until)*). Therefore we find it useful to use one type, PREP, for all "edat" errors. "Edat" as suffix is classified with morphological analysis. Word level PREP, DET, CONJ and PRON categories are simply classified with the help of POS tags and a predefined vocabulary for each type. MORPH category is too wide to cover any error type in Turkish. Therefore, we decide to discard this category and distribute its coverage to other, mainly :INFL, sub-categories.

In order to catch morphological errors, we need morphological analysis and POS tags of the words. "NOUN", "ADJ" and "ADV" tags are the main concerns as they may be derived from either a verb stem or a noun stem. The nouns, adjectives and adverbs that are derived from a verb (Infinitive, Participle and Gerund) may have the same suffixes as the ones inflected from a noun. ADJ, ADV and NOUN categories are assigned if the correct and erroneous tokens' lemmas are different but their morphological properties are the same as it means that the choice of word is wrong but the inflections are correct. The sub-categories (:INFL, :POSS, :NUM) are assigned if the word lemma is the same for both correct and erroneous tokens but possessive, numeral or other inflections are wrong.

VERB error types are classified similar to NOUN types. If the inflection is the same for both correct and erroneous verbs but the lemmas are different, this means that the choice of verb is wrong.

The :SVA sub-category is detected if the correct and erroneous verb contains different *personal suffixes*. Though it might be an inflection error as the error may be caused by the inflection inability rather than the wrong choice of "personal suffix". The :TENSE sub-category is assigned if the verb lemmas are the same but the chosen tense is wrong although the produced word is a valid verb. The :INFL sub-category is the other inflection errors that do not fall into neither :TENSE nor :SVA.

4 Experimental Results

4.1 Dataset

We collect 106 erroneous and corrected sentences [3] from academic studies discussing the errors made by foreign learners of Turkish; İltar (2021), Çelik (2019), Altintop (2018) and Dizeli and Sonkaya (2021). The mentioned studies categorizes some of the sentences under error types different from Table 1; e.g., diacritics usage errors, usage of dialects in writings, wrong usage of noun cases and wrong usage of noun number suffix. We map these types to the closest ERRANT types such as SPELL, NOUN:INFL.

In order to evaluate the proposed system, the error types have been reviewed by another linguist and the edits are labeled according to the discussed error types. The distribution and the number of error types can be seen in Table-2. It can be seen that the inflection sub-categories (:INFL) have much more samples than other morphological error types. This is due to the inflectional richness of Turkish and there are many sub-categories under :INFL.

4.2 Evaluation

We use M^2 file format (Dahlmeier and Ng, 2012) and measure the system performance using ERRANT's default scorer to compare the system output and the gold reference. The default scorer calculates span-based correction precision, recall and $F_{0.5}$ scores between two annotated M^2 files (Bryant et al., 2019).

We evaluate ERRANT-TR's error type classifier on manually annotated 106 parallel sentences. We consider the edit labels, which were reviewed by a linguist, as the ground truth and compare them to the system's output. ERRANT-TR achieves an average 77.04% $F_{0.5}$ score of span-based correction score. In order to better understand the system's strengths and weaknesses, we provide the $F_{0.5}$ scores per error type which can be seen in Table 2. Some inflection errors are classified as SPELL due to limited morphological analysis of an erroneous token. However, the overall $F_{0.5}$ score of the classifier is 77.04%. There are not many studies which compares their ERRANT implementation with gold standard data as we do. Only Korre et al. (2021) measured it this way and reported a maximum $F_{0.5}$ score of 43.50% on one dataset and 86.28% on another.

[3]The collected dataset is publicly available from https://github.com/harunuz/erranttr.git

5 Discussion

In this work, we developed the first error annotation tool for Turkish using the error types introduced in ERRANT. Even though the main purpose of these types is meant to cover the most common error types in a parallel corpus, during the development of ERRANT-TR and the annotation of the validation dataset, we observed that they are not completely applicable to agglutinative languages like Turkish. Some common errors in Turkish are not exactly covered with these types and some error types (e.g. MORPH, VERB:INFL) are too wide. Especially the morphological ones need to be expanded to cover a wide range of inflectional and derivational errors. On the other hand, the advantage of an agglutinative language is that the suffixes are usually added to a word stem in an order (see Good and Alan (1999) and Part 2 of Göksel and Kerslake (2004) for the case of Turkish). This phenomenon helps to classify error types into hierarchical classes and makes it relatively easier to implement a decision making algorithm.

Bryant et al. (2017) proposed a hierarchical relationship between error types. For instance, a NOUN:POSS error is also a NOUN error or a VERB:TENSE error is also a VERB error. Knowing these relationships prior to developing a classifier will help in classifying sub-types and will provide more information during the test phase. In Turkish, even more detailed hierarchical relationship between error types can be established due to the rich derivational morphology. To further illustrate this, we provide a simple example:

```
yap (VERB)
    -tık(ğ) (PastPart+A3sg+Noun)
        -ın (P2sg)
            -ı (Acc)
```

The verb lemma *yap-* has the following transformations: it is derived to a participle (an adjective-verb); the modified noun (third person singular) is dropped and the participle becomes a noun; then it is inflected with a second person singular possessive; lastly it is inflected with an accusative case.

Error Type	Number of Occurrence	Percentage of Occurrence (%)	P	R	$F_{0.5}$
ADJ	2	1.02	0.4	1.0	0.45
ADJ:FORM	0	0	-	-	-
ADV	5	2.55	1.0	0.8	0.95
CONJ	1	0.51	1.0	1.0	1.0
CONTR	0	0	-	-	-
DET	0	0	-	-	-
MORPH	0	0	-	-	-
NOUN	1	0.51	0.1	1.0	0.12
NOUN:INFL	43	21.93	0.88	0.86	0.87
NOUN:NUM	20	10.20	0.73	0.95	0.76
NOUN:POSS	17	8.67	0.87	0.41	0.71
ORTH	10	5.10	1.0	0.9	0.97
OTHER	16	8.16	0.52	0.68	0.55
PART	0	0	-	-	-
PREP	3	1.53	1.0	1.0	1.0
PRON	0	0	-	-	-
PUNCT	6	3.06	1.0	0.83	0.96
SPELL	38	19.38	0.82	0.73	0.80
UNK	0	0	-	-	-
VERB	8	4.08	0.75	0.75	0.75
VERB:FORM	0	0	-	-	-
VERB:INFL	16	8.16	0.6	0.27	0.48
VERB:SVA	3	1.53	1.0	1.0	1.0
VERB:TENSE	5	2.55	1.0	1.0	1.0
WO	2	1.02	1.0	1.0	1.0
Total / Micro Average	196	$\sim$100.00	0.77	0.77	0.77

Table 2: (Left) Error code, the number of occurrences and the percentage in the dataset. Note that each sentence may have more than one error. (Right) Precision, recall and $F_{0.5}$ scores of ERRANT-TR on the dataset for each error type.

Let us suppose that the learner made an error in possessive and used a third person singular possessive suffix *"-ı"*. Classifying this error as a NOUN:POSS type loses a valuable information of a common possessive error case in *participles* (İltar, 2021). Moreover, knowing that a verb can not be inflected with a possessive suffix, even though the lemma of the word is a verb we can safely discard the VERB and its sub-types while classifying this error. Therefore, a more precise classification system and an improved labeling scheme can be created by establishing hierarchical relationships among error types in a more intricate manner.

For nouns and verbs, "INFL" sub-categories mostly cover other sub-categories. For example, a "NOUN:NUM" or a "NOUN:POSS" error in Turkish can also be considered a "NOUN:INFL" error as the *number* and *possessive* properties are provided with inflectional suffixes. Furthermore, in the specific case of Turkish, a possessive error can also be considered a *genitive construction* error. As can be seen in the example below, a possession suffix *-ı* is appended to the head (modified noun) in a genitive construction and the modifier is appended with a *tamlayan (modifier)* suffix *-ın*.

```
Kitabın kapağı -> Kitab-ın kapağ-ı

(The book's cover)
```

Despite technically being a possession, errors in this type of phrase may also fall under the category of "genitive construction" errors due to frequent inflectional errors made by non-native Turkish learners in such phrases, even though they use possessives correctly in other contexts

5.1 Uncovered Turkish Specific Cases

Although it is not a grammatical error, the usage of Turkish specific characters (diacritics) could be non trivial for some learners. In addition to that, specifically on social media, some people choose to write with ASCII characters rather than their Turkish correspondents. The usage of *dialects* in written text is also the result of either a genuine mistake or a preference (Eryiğit and Torunoğlu-Selamet, 2017). Therefore, both *accent* and *diacritics* error types might be considered to take into account while developing Turkish GEC systems.

In Turkish, passivization, reciprocal verbs and the meaning of *making somebody do something* are all done with inflectional suffixes that are appended to a verb. The errors on these inflections are common enough to be considered standalone error categories.

6 Future Work

In this section, we aim to address the issues concerning the labeling and classification of grammatical error types that we have discussed earlier. We also outline possible areas for further research and suggest potential enhancements for ERRANT-TR.

Categorizing certain errors based solely on morphological analysis can be difficult, particularly when dealing with multi-token edits. Nonetheless, supplementing morphological analysis with dependency parsing can assist in precisely aligning and categorizing multi-token errors. Our intention is to use these techniques to enhance the system's performance on existing error types and address the Turkish-specific errors discussed in Section-5.1.

The evaluation of an automatic annotation toolkit (detection, alignment and classification capabilities) is a time and resource consuming process. One needs a big amount of already-annotated parallel data with high inter-annotator agreement. As we did not possess such data we only evaluated the classifier. In this work, the ground truth data is considered properly aligned and corrected. Therefore, the evaluation process has still room for improvement. We plan to test the system on a real world learner corpus which is being collected at the moment as part of an ongoing research.

Error type classification relies on the accuracy of the morphological analysis and disambiguation. Therefore, a potential mistake in these steps may yield incorrect classifications. In order to improve the system in the future, a better morphological analyzer can be used. There is also a room for improvement in the decision making pipeline of the classifier. The detection of certain error types is not trivial with only the morphological analysis as discussed earlier.

7 Conclusion

In this paper, we introduced ERRANT-TR a grammatical error annotation and automatic evaluation toolkit for Turkish. It automatically annotates the error types in a parallel corpus. We designed a decision making pipeline for Turkish based on morphological analysis information and hand-crafted specific vocabularies (for CONJ, PREP, PRON, DET types). We discussed and proposed potential changes to the error categories (which have been introduced mainly for English) in order to cover inflectional and derivational properties of Turkish, an agglutinative language.

We created a small evaluation dataset consisting of 106 erroneous-corrected sentence pairs collected from academic studies. ERRANT-TR achieves an average 77.04% $F_{0.5}$ score on this dataset. We discussed the strengths and weaknesses of the system based on this evaluation. In the future, we will evaluate and improve the system on the first Turkish learner corpus.

ERRANT-TR will also help learners and teachers by being used as a semi-automatic annotation toolkit while annotating erroneous-correct sentence pairs and reduce the time required to create parallel corpora drastically.

Limitations

The used morphological analyzer does not provide morpheme and POS tag lists compatible with Universal Postags (de Marneffe et al., 2021). Therefore, there might be issues with adapting this work in other languages while using the system as is.

Although the computational power requirements for the system are low, it does not work well with parallel computing. Thus a large amount of data might take long time to be processed.

Lastly, the proposed classifier system has been developed and evaluated with publicly available, limited samples from academic studies in linguistics. The data does not represent real world scenarios well enough. Therefore, we will test the system on real data only after the first Turkish learner corpus (which is mentioned at the end of the Section 5) is available.

References

Ahmet Afsin Akın and Mehmet Dündar Akın. 2007. Zemberek, an open source NLP framework for Turkic languages. *Structure*, 10(2007):1–5.

Esra Altintop. 2018. Yabancı dil olarak Türkçe öğrenen öğrencilerin kompozisyonlarındaki ek yanlışları ve nedenleri.

Ugurcan Arikan, Onur Gungor, and Suzan Uskudarli. 2019. Detecting clitics related orthographic errors in Turkish. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 71–76, Varna, Bulgaria. INCOMA Ltd.

B. Erdem Dağıstanlıoğlu Başak Karakoç Öztürk. 2017. Yabancı dil olarak Türkçe öğrenen öğrencilerin yazdıkları öyküleyici metinlerdeki yazım sorunları: Bir hata analizi örneği. *Uluslararası 2. Sosyal Bilimler Sempozyumu (Asos Congress)*.

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 *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. 2022. Grammatical error correction: A survey of the state of the art. *arXiv preprint arXiv:2211.05166*.

Aslı Çelik. 2019. İkinci/yabancı dil olarak Türkçe öğrenenlerin sözel hatalarına ilişkin öğretici tercihleri ve tutumları.

Jhih-Jie Chen, Yi-Dong Wu, Yu-Chuan Tai, Ching-Yu Yang, Hai-Lun Tu, and Jason S. Chang. 2019. Extracting grammatical error corrections from wikipedia revision history. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 6016–6018.

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.

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, Atlanta, Georgia. Association for Computational Linguistics.

Sam Davidson, Aaron Yamada, Paloma Fernandez Mira, Agustina Carando, Claudia H. Sanchez Gutierrez, and Kenji Sagae. 2020. Developing NLP tools with a new corpus of learner Spanish. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 7238–7243, Marseille, France. European Language Resources Association.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308.

Merve Dizeli and Zeynep Zeliha Sonkaya. 2021. Birinci ve ikinci yabancı dil olarak Türkçe öğrenen öğrencilerin yaptıkları dilbilimsel hatalar. *Türkiye Sosyal Araştırmalar Dergisi*, 25(1):225 – 234.

Gülşen Eryiğit and Dilara Torunoğlu-Selamet. 2017. Social media text normalization for Turkish. *Natural Language Engineering*, 23:1–41.

Mariano Felice, Christopher Bryant, and Ted Briscoe. 2016. Automatic extraction of learner errors in ESL sentences using linguistically enhanced alignments. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 825–835, Osaka, Japan. The COLING 2016 Organizing Committee.

Mehmet Fidan. 2019. Türkçe öğrenen yabancı öğrencilerin yazılı anlatım metinlerinin yazım ve noktalama kuralları yönünden değerlendirilmesi. *EKEV Akademi Dergisi*, 0(77):253 – 266.

Aslı Göksel and Celia Kerslake. 2004. *Turkish: A comprehensive grammar*. Routledge.

Jeff Good and CL Alan. 1999. Affix-placement variation in Turkish. In *Annual Meeting of the Berkeley Linguistics Society*, volume 25, pages 63–74.

Roman Grundkiewicz and Marcin Junczys-Dowmunt. 2014. The wiked error corpus: A corpus of corrective wikipedia edits and its application to grammatical error correction. In *Advances in Natural Language Processing*, pages 478–490, Cham. Springer International Publishing.

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.

Masahiro Kaneko, Masato Mita, Shun Kiyono, Jun Suzuki, and Kentaro Inui. 2020. Encoder-decoder models can benefit from pre-trained masked language models in grammatical error correction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4248–4254, Online. Association for Computational Linguistics.

Anisia Katinskaia, Maria Lebedeva, Jue Hou, and Roman Yangarber. 2022. Semi-automatically annotated learner corpus for Russian. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 832–839, Marseille, France. European Language Resources Association.

Shun Kiyono, Jun Suzuki, Masato Mita, Tomoya Mizumoto, and Kentaro Inui. 2019. An empirical study of incorporating pseudo data into grammatical error correction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1236–1242, Hong Kong, China. Association for Computational Linguistics.

Katerina Korre, Marita Chatzipanagiotou, and John Pavlopoulos. 2021. ELERRANT: Automatic grammatical error type classification for Greek. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 708–717, Held Online. INCOMA Ltd.

Berker Kurt. 2020. Yabancı dil olarak Türkçe öğrenenlerin konuşmada yaptıkları dilsel hatalar ve iletişim stratejileri üzerine fenomenolojik bir çalışma. *International Journal of Languages Education*, 8.1:51–70.

Jared Lichtarge, Chris Alberti, Shankar Kumar, Noam Shazeer, Niki Parmar, and Simon Tong. 2019. Corpora generation for grammatical error correction. 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 3291–3301, Minneapolis, Minnesota. Association for Computational Linguistics.

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.

Jakub Náplava, Milan Straka, Jana Straková, and Alexandr Rosen. 2022. Czech Grammar Error Correction with a Large and Diverse Corpus. *Transactions of the Association for Computational Linguistics*, 10:452–467.

Kostiantyn Omelianchuk, Vitaliy Atrasevych, Artem Chernodub, and Oleksandr Skurzhanskyi. 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.

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.

Esin Yağmur Şahin. 2013. Yabancı dil olarak Türkçe öğrenen öğrencilerin yazılı anlatımlarındaki ek yanlışları. *Tarih Okulu Dergisi*, 2013(XV).

Yu Wang, Yuelin Wang, Jie Liu, and Zhuo Liu. 2020. A comprehensive survey of grammar error correction. *arXiv preprint arXiv:2005.06600*.

Soyoung Yoon, Sungjoon Park, Gyuwan Kim, Junhee Cho, Kihyo Park, Gyu Tae Kim, Minjoon Seo, and Alice Oh. 2022. Towards standardizing Korean grammatical error correction: Datasets and annotation.

Wei Zhao, Liang Wang, Kewei Shen, Ruoyu Jia, and Jingming Liu. 2019. Improving grammatical error correction via pre-training a copy-augmented architecture with unlabeled data. 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 156–165, Minneapolis, Minnesota. Association for Computational Linguistics.

Latif İltar. 2021. Türkçeyi yabancı dil olarak öğrenenlerin C1 yazma sınavındaki dil bilgisi hatalarının düzeylere göre dağılımı. *Bayburt Eğitim Fakültesi Dergisi*, 16(Özel Sayı):129 – 147.