Towards SMT-Assisted Error Annotation of Learner Corpora

Nadezda Okinina
Eurac Research
viale Druso 1, Bolzano, Italy
nadezda.okinina@eurac.edu

Lionel Nicolas
Eurac Research
viale Druso 1, Bolzano, Italy
lionel.nicolas@eurac.edu

Abstract

English. We present the results of prototypical experiments conducted with the goal of designing a machine translation (MT) based system that assists the annotators of learner corpora in performing orthographic error annotation. When an annotator marks a span of text as erroneous, the system suggests a correction for the marked error. The presented experiments rely on word-level and character-level Statistical Machine Translation (SMT) systems.

Italian. *Presentiamo i risultati degli esperimenti prototipici condotti con lo scopo di creare un sistema basato sulla traduzione automatica (MT) che assista gli annotatori dei corpora degli apprendenti di lingue durante il processo di annotazione degli errori ortografici. Quando un annotatore segna un segmento di testo come errato il sistema suggerisce una correzione dell'errore segnato. Gli esperimenti presentati utilizzano dei sistemi statistici di traduzione automatica (SMT) al livello di parole e di caratteri.*

1 Introduction

Manual error annotation of learner corpora is a time-consuming process which is often a bottleneck in learner corpora research. *"Computer learner corpora are electronic collections of authentic FL/SL textual data assembled according to explicit design criteria for a particular SLA/FLT*[1] *purpose. They are encoded in a standardised and homogeneous way and documented as to their origin and provenance"* (Granger, 2002). Error-annotated learner corpora serve the needs of language acquisition studies and pedagogy development as well as help the creation of natural language processing tools such as automatic language proficiency level checking systems (Hasan et al., 2008) or automatic error detection and correction systems (see Section 2). In this paper we present our first attempts at creating a system that would assist annotators in performing orthographic error annotation by suggesting a correction for specific spans of text selected and marked as erroneous by the annotators. In the prototypical experiments, the suggestions are generated by word-level and character-level SMT systems.

This paper is organized as follows: we review existing approaches to automatic error correction (Section 2), introduce our experiments (Section 3), present the data we used (Section 4), describe and discuss the performed experiments (Section 5) and conclude the paper (Section 6).

2 Related Work

Orthographic errors are mistakes in spelling, hyphenation, capitalisation and word-breaks (Abel et al., 2016). Automatic orthographic error correction can benefit from methods recently developed for grammatical error correction (GEC) such as methods relying on SMT and Neural Machine Translation (NMT) (Chollampatt et al., 2017, Ji et al., 2017, Junczys-Dowmunt et al., 2016, Napoles et al., 2017, Sakaguchi et al., 2017, Schmaltz et al., 2017, Yuan et al., 2016 etc.). These approaches treat error correction as a MT task from incorrect to correct language. In the case of orthographic error correction these "languages" are extremely close, which greatly facilitates the MT task. In that aspect, error correction is similar to the task of translating closely-related languages such as, for example, Mace-

[1] FL: foreign language, SL: second language, SLA: second language acquisition, FLT: foreign language teaching

donian and Bulgarian (Nakov et al., 2012). In our experiments, we rely on the implementation of SMT models provided by the Moses toolkit (Koehn et al., 2007).

SMT and NMT can be easily adapted to new languages, but their performance depends on the amount and quality of the training data. In order to make up for lack of parallel corpora of texts containing language errors and their correct equivalents, various techniques for resource construction have been suggested, such as using the World Wide Web as a corpus (Whitelaw et al., 2009), parsing corrective Wikipedia edits (Grundkiewicz et al., 2014) or injecting errors in error-free text (Ehsan et al., 2013). For our prototypical experiments, we deliberately limit ourselves to the manually-curated high-quality data at our disposal and use existing German error-annotated corpora as training data.

In recent years learner corpora of German have been used for the creation of systems for automatic German children's spelling errors correction (Stüker et al., 2011, Laarmann-Quante, 2017), but no work has been done on automatic orthographic error correction of adult learner texts.

3 Objectives of the Experiments

The particularity of our work is that we focus on a specific use-case where annotators are assisted in error-tagging newly created learner corpora. To ensure the relevance of our system and limit false positives that would hinder its adoption, the targeted use-case is to only suggest corrections while leaving the task of selecting the error to the linguist. Aforementioned GEC systems take as input text containing language errors and produce corrected text. Thus, they may introduce changes in any part of the text, even where no errors are observed. In order to prevent such behavior, we only submit to our system spans of text marked as erroneous by annotators, while leaving out spans of text not containing errors. Therefore, our system is not directly comparable to existing GEC systems.

A given language error may have more than one possible correction, but in the presented research we limit ourselves to orthographic errors that in most cases have only one correction (Nerius et al., 2007). Our system is meant to be used for the creation of new learner corpora in the Institute for Applied Linguistics where learner corpora of German, Italian and English are created and studied (Abel et al., 2013, Abel et al., 2015, Abel et al., 2016, Abel et al., 2017, Zanasi et al., 2018).

Preliminary experiments with the freely available vocabulary-based spell checking tool Hunspell[2] yielded unsatisfactory results (see Section 5.1) and incited us to try SMT in order to train an error-correction system and tune it to the specific nature of our data. We thus performed a series of experiments to perform a preliminary evaluation of the range of performances of different n-gram models when trained on small-scale data (Section 5.1), studied the impact of the similarity between training data and test data to understand which datasets are the most optimal to train our models on (Sections 5.2 and 5.3) and finally made preliminary attempts to improve the performance by optimising the usage of the SMT systems (Section 5.4).

As our systems are not directly comparable to GEC systems, the usual metrics used to evaluate GEC systems are not fully adequate, because they target a similar but different use case. We thus evaluate our systems according to their accuracy that we define as a ratio between the number of suggestions matching the target hypothesis present in the test data (TH)[3] and the whole number of annotated errors. However, accuracy is not the only criteria as it is also important not to disturb the annotators with irrelevant suggestions: it is better not to suggest any TH than to suggest a wrong one. In order to control the ratio between right and wrong suggestions, we also evaluate our systems according to their precision. We define precision as a ratio between the number of suggestions matching the TH and the whole number of suggestions, correct and incorrect, thus excluding the errors for which the system was consulted, but no correction was suggested. Precision is mainly used as a quality threshold which should remain high, whereas our main performance measure is accuracy.

4 Corpora Used

Our experiments rely on three error-annotated learner corpora: KoKo, Falko and MERLIN.

KoKo is a corpus of 1.503 argumentative essays (811.330 tokens) of written German L1[4] from high school pupils, 83% of which are native speakers of German (Abel et al., 2016). It relies

[2] http://hunspell.github.io/

[3] The TH corresponds to a correction associated with each error (Reznicek et al., 2013).

[4] first language, native language

on a very precise error annotation scheme with 29 types of orthographic errors.

The Falko corpus consists of six subcorpora (Reznicek et al., 2012) out of which we are using the subcorpus of 107 error-annotated written texts by advanced learners of L2 [5] German (122.791 tokens).

The MERLIN corpus was compiled from standardized, CEFR[6]-related tests of L2 German, Italian and Czech (Boyd et al., 2014). We are using the German part of MERLIN that contains 1033 learner texts (154.335 tokens): a little bit more than 200 texts for each of the covered CEFR levels (A1, A2, B1, B2, and C1).

Due to the differences in content and format, we do not use all three learner corpora in all the experiments. KoKo is our main corpus, because of its larger size, easy to use format and detailed orthographic error annotation. We use it in training, validation and testing of our SMT systems. Falko is smaller and its format does not allow an easy alignment of orthographic errors, we thus only use it in some experiments as part of the training corpus (Sections 5.1 and 5.2). MERLIN was annotated similarly to KoKo, therefore error-correction results obtained for these two corpora are easily comparable. Furthermore, MERLIN is representative of different levels of language mastery. We thus use it for testing some of our systems (Section 5.2).

As the language model for our character-based SMT systems cannot be generated from the limited amount of data provided by learner corpora, for that purpose we used 3.000.000 sentences of a German news subcorpus from the Leipzig Corpora Collection[7].

5 Prototypical Experiments

5.1 Testing Different N-Gram Models

We started by testing SMT word and character-based language models with various numbers of n-grams in order to understand which one could suffer less from data scarcity and thus best suit our data[8] (Table 1). We used Moses default values for all the other parameters. The systems were trained on a parallel corpus composed of learner texts and their corrected versions from Falko and KoKo. In each fold of the 10-fold validation, 1/10 of KoKo is taken out of the training corpus and used as a validation corpus.

Since our objective was to only observe the overall adequateness of the SMT models, we only attempted to optimise the way the SMT models were used at a later stage (see Section 5.4).

These prototypical experiments showed that all the SMT models have a rather high precision and that, for this amount of training data, the SMT model that performed best is the word 5-gram model. It yielded an encouraging result of 39% of accuracy and 89% of precision, which is far better than the 11% of accuracy and 8% of precision originally obtained with Hunspell. However, 39% of accuracy were obtained by training on Falko and 9/10 of KoKo and validating on 1/10 of KoKo, which would be the configuration we would have towards the end of the annotation of a new learner corpus. We thus proceeded with our experiments by testing how the SMT models would perform at an earlier stage.

	word-grams				character-grams		
	1	3	5	10	6	10	15
Prec.	84%	87%	**89%**	84%	83%	86%	87%
Acc.	32%	37%	**39%**	38%	16%	21%	29%

Table 1: 10-fold validation on KoKo of SMT models trained on KoKo and Falko.

5.2 Testing the Models on New Data

At an early stage of the annotation of a new learner corpus, an error-correction system could be trained on an already existing corpus. We thus tried to apply the different models trained on Falko, KoKo and the newspapers to MERLIN. However, none of the 7 models presented in the previous section achieved more than 13% of accuracy and 70% of precision on the whole MERLIN corpus. Despite that, these experiments highlighted an interesting aspect: all the models performed better on MERLIN texts of higher CEFR levels compared to MERLIN texts of lower CEFR levels (Table 2). We suspect this phenomenon to be due to the fact that the level of language mastery of MERLIN texts of higher CEFR levels is closer to the level of language mastery of KoKo and Falko texts. This observation indicates that the training and test data must attest to the same level of language mastery, because mistakes made by beginner language learners tend to differ noticeably from mistakes made by advanced language learners. Therefore,

[5] second language, foreign language

[6] Common European Framework of Reference for Languages

[7] http://hdl.handle.net/11022/0000-0000-2417-E

[8] The computational results presented have been achieved in part using the Vienna Scientific Cluster (VSC).

using existing learner corpora as training data is a difficult task as most of them target different types of learners with different profiles and bias towards specific kinds of errors.

	A1	A2	B1	B2	C1
Prec.	60%	61%	77%	72%	78%
Acc.	15%	9%	12%	14%	17%

Table 2: precision and accuracy of the word 5-gram model trained on KoKo and Falko when tested on MERLIN texts of different CEFR levels.

5.3 Training and Testing on One Corpus

The results of the previous experiments incited us to train an SMT model on a small part of a corpus and test it on a bigger part of the same corpus in order to observe how an SMT model would behave when trained on an already annotated part of a new learner corpus. We thus performed 3-fold validation experiments with a word 5-gram model taking 1/3 of KoKo as training data and 2/3 of KoKo as test data and obtained 30% of accuracy[9]. This result was much better than 13% of accuracy we had obtained by training SMT systems on KoKo and Falko and testing them on MERLIN. We thus decided to pursue our experiments with KoKo as both training and test data.

In order to observe the evolution of the system's performance with the growth of the corpus, we also trained it on 2/3 of KoKo and tested it on 1/3 of KoKo. Augmenting the training corpus size did not change the system's performance (Table 3, line 1). Such results tend to indicate that most of the performance can be obtained at an earlier stage of the annotation process.

5.4 Improving the Performance

After evaluating the impact of the training data on the system's performance, we switched our focus to the optimisation of the way SMT models were used. First of all, we tried to take into account not only the highest-ranked suggestion of Moses, that in many cases was equal to the error text (i.e. no correction was suggested), but also the lower-ranked suggestions in order to find the highest-ranked suggestion that was different from the error text. This change considerably improved the accuracy for both corpus sizes and only slightly deteriorated the precision (Table 3, line 2).

In order to further improve the performance, we decided to combine the word-based and character-based systems. For this first experiment we chose the best-performing of the word-based systems which is the word 5-gram model and the second best performing of the character-based systems which is the character 10-gram model. We chose the character 10-gram model for practical reasons: it is considerably less resource-consuming than the character 15-gram model. By applying both the word 5-gram and the character 10-gram models to the same data and comparing the overlap in their responses, we verified their degree of complementarity. This experiment showed that only in 18% of cases the word-based and character-based models both suggest a correction (corresponding or not to the TH). In 39% of cases only the word-based system suggests a correction and in 5% of cases only the character-based system suggests a correction. It means that by combining the two systems it is possible to improve the overall performance. We calculated the maximum theoretical accuracy[10] of such a combined system and came to a conclusion that it cannot exceed 53% when trained on 1/3 of KoKo and 60% when trained on 2/3 of KoKo (Table 3, line 3).

By simply giving preference to the word-based model before consulting the character-based model, we almost achieved the maximum theoretical accuracy (Table 3, line 4).

However, we realised that by augmenting the training corpus size, we augmented the accuracy, but slightly deteriorated the precision.

By analysing the performance of different modules (word 5-gram highest-ranked suggestions, word 5-gram lower-ranked suggestions, character 10-gram) on different kinds of errors, we could observe that their performance differs according to types of errors. For example, the lower-ranked suggestions of the word-based model introduce a lot of mistakes in the correction of errors where one word was erroneously written as two separate words (e.g. *Sommer fest* instead

[9]We also calculated the BLEU score for this model and obtained 95%. This result shows that the BLEU score is irrelevant for the evaluation of error correction systems such as ours that cannot introduce errors in error-free spans of text.

[10]The maximum theoretical accuracy would be achieved if it was possible to always choose the right system to consult for each precise error (word-based or character-based) and never consult the system that gave a wrong result when the other system gave a correct result. In that case the maximum potential of both systems would be used.

of *Sommerfest*). We tried to prevent such false corrections by not consulting the lower-ranked suggestions of the word-based model for errors containing spaces. By introducing this rule we succeeded in improving the precision at the cost of loosing some accuracy (Table 3, line 5). This experiment showed that add-hoc rules might not be a workable solution and a more sophisticated approach should be considered if we intend to dynamically combine several systems. In order to obtain better results combining two or more word-based and character-based systems, further experiments should be conducted.

		train. 1/3 valid. 2/3	train. 2/3 valid. 1/3
1	word highest-ranked corr.	30% (88%)	30% (88%)
2	word lower-ranked corr.	48% (84%)	55% (83%)
3	max. theoretical accuracy word lower-ranked + character	53% (85%)	60% (84%)
4	word lower-ranked + character	53% (84%)	59% (83%)
5	word lower-ranked +character with rule on spaces	52% (88%)	57% (88%)

Table 3: accuracy and precision (in brackets) of different systems according to training corpus size (3-fold validation on KoKo).

6 Conclusion

Our preliminary experiments brought us to the conclusion that a SMT system trained on a manually annotated part of a learner corpus can be helpful in error-tagging the remaining part of the same learner corpus: it is possible to train a system that would propose the right correction for half of the orthographic errors outlined by the annotators while proposing very few wrong corrections. Such results are satisfactory enough to start integrating the system into the annotation tool we use to create learner corpora (Okinina et al., 2018).

The combination of a word-based and a character-based systems gave promising results, therefore we intend to continue experimenting with multiple combinations of word-based and character-based systems. We are also considering the possibility to rely on other technologies (Bryant, 2018). As in our experiments we only wanted to observe the range of performances we could expect, we trained our models with the default configuration provided with the MOSES toolkit and did not perform any tuning of the parameters. Future efforts will focus on evaluating how relevant the tuning of parameters can be for such a MT task.

The choice of training data for our experiments was dictated by the availability of high-quality resources. In future experiments we would like to enlarge the spectrum of resources considered for our experiments and work with other languages, in particular with Italian and English.

Acknowledgements

We would like to thank the reviewers as well as our colleagues Verena Lyding and Alexander König for their useful feedback and comments.

References

Abel, A., Konecny, C., Autelli, E.: Annotation and error analysis of formulaic sequences in an L2 learner corpus of Italian, *Third International Learner Corpus Research Conference*, 2015, Book of abstracts, pp. 12-15.

Abel, A., Glaznieks, A., Nicolas, L., Stemle, E.: An extended version of the KoKo German L1 Learner corpus, *Proceedings of the Third Italian Conference on Computational Linguistics CliC-it*, Naples, Italy, 2016, pp. 13-18.

Abel, A., Glaznieks, A.: „Ich weiß zwar nicht, was mich noch erwartet, doch ...“ – Der Einsatz von Korpora zur Analyse textspezifischer Konstruktionen des konzessiven Argumentierens bei Schreibnovizen, *Corpora in specialized communication*, vol. 4, Bergamo, 2013, pp. 101-132.

Abel, A., Vettori, C., Wisniewski, K.: KOLIPSI. Gli studenti altoatesini e la seconda lingua: indagine linguistica e psicosociale, vol. 2, Eurac Research, 2017.

Boyd, A., Hana, J., Nicolas, L., Meurers, D., Wisniewski, K., Abel, A., Schöne, K., Štindlová, B., Vettori, C.: The MERLIN corpus: Learner language and the CEFR, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC),* 2014, pp. 1281-1288.

Bryant, C.: Language Model Based Grammatical Error Correction without Annotated Training Data, *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, 2018, pp. 247–253.

Chollampatt, S., Ng, H.: Connecting the Dots: Towards Human-Level Grammatical Error Correction, *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, 2017, pp. 327-333.

Granger, S.: A Bird’s Eye View of Learner Corpus Research. In Granger, S., Hung, J., Petch-Tyson, S.

(eds.), Computer Learner Corpora, Second Language Acquisition and Foreign Language Teaching, Amsterdam & Philadelphia: Benjamins, 2002, pp. 3-33.

Ehsan, N., Faili, H.: Grammatical and context-sensitive error correction using a statistical machine translation framework, *Software – Practice and Experience*, 2013, 43, pp. 187-206.

Grundkiewicz, R., Junczys-Dowmunt, M.: The WikEd Error Corpus: A Corpus of Corrective Wikipedia Edits and Its Application to Grammatical Error Correction. In Przepiórkowski, A., Ogrodniczuk, M. (eds.), Advances in Natural Language Processing. NLP 2014. Lecture Notes in Computer Science, vol. 8686. Springer, Cham, 2014, pp. 478-490.

Hasan, M. M., Khaing, H. O.: Learner Corpus and its Application to Automatic Level Checking using Machine Learning Algorithms, *Proceedings of ECTI-CON*, 2008, pp. 25-28.

Ji, J., Wang, Q., Toutanova, K., Gong, Y., Truong, S., Gao, J.: A Nested Attention Neural Hybrid Model for Grammatical Error Correction, ArXiv e-prints, 2017.

Junczys-Dowmunt, M., Grundkiewicz, R.: Phrase based machine translation is state-of-the-art for automatic grammatical error correction, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics*, Austin, Texas, 2016, pp. 1546–1556.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., Herbst, E.: Moses: Open source toolkit for statistical machine translation, *Proceedings of ACL '07*, Prague, Czech Republic, 2007, pp. 177–180.

Laarmann-Quante, R.: Towards a Tool for Automatic Spelling Error Analysis and Feedback Generation for Freely Written German Texts Produced by Primary School Children, *Proceedings of the Seventh ISCA workshop on Speech and Language Technology in Education*, 2017, pp. 36-41.

Nakov, P., Tiedemann, J.: Combining Word-Level and Character-Level Models for Machine Translation Between Closely-Related Languages, *Proceedings of the 50th Annual Meeting of the Association of Computational Linguistics (ACL)*, 2012, pp. 301-305.

Napoles, C., Sakaguchi, K., Tetreault, J.: JFLEG: A Fluency Corpus and Benchmark for Grammatical Error Corrections, *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics,* vol. 2, Short Papers. Association for Computational Linguistics, Valencia, Spain, 2017, pp. 229–234.

Nerius, D. et al.: *Deutsche Orthographie*. 4., neu bearbeitete Auflage. Hildesheim/Zürich/New York: Olms Verlag, 2007.

Okinina, N., Nicolas, L., Lyding, V.: Transc&Anno: A Graphical Tool for the Transcription and On-the-Fly Annotation of Handwritten Documents, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC)*, 2018, pp. 701-705.

Reznicek, M., Lüdeling, A., Hirschmann, H.: Competing Target Hypotheses in the Falko Corpus: A Flexible Multi-Layer Corpus Architecture, *Automatic Treatment and Analysis of Learner Corpus Data*, John Benjamins Publishing Company, Amsterdam/Philadelphia, 2013, pp. 101-123.

Reznicek, M., Lüdeling, A., Krummes, C., Schwantuschke, F.: Das Falko-Handbuch Korpusaufbau und Annotationen, Version 2.0, 2012.

Sakaguchi, K., Post, M., Van Durme, B.: Grammatical Error Correction with Neural Reinforcement Learning, *Proceedings of the Eighth International Joint Conference on Natural Language Processing*, Asian Federation of Natural Language Processing, Taipei, Taiwan, pp. 366–372.

Schmaltz, A., Kim, Y., Rush, A., Shieber, S.: Adapting Sequence Models for Sentence Correction, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 2807-2813.

Stüker S., Fay, J., Berkling, K.: Towards Context-dependent Phonetic Spelling Error Correction in Children's Freely Composed Text for Diagnostic and Pedagogical Purposes, *Interspeech*, 2011.

Whitelaw, C., Hutchinson, B., Chung, G., Ellis, G.: Using the Web for Language Independent Spellchecking and Autocorrection, *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, Singapore, 2009, pp. 890-899.

Yuan, Z., Briscoe, T.,: Grammatical Error Correction Using Neural Machine Translation, *Proceedings of NAACL-HLT 2016*, 2016, pp. 380-386.

Zanasi, L., Stopfner, M.: Rilevare, osservare, consultare. Metodi e strumenti per l'analisi del plurilinguismo nella scuola secondaria di primo grado. In Coonan, C., Bier, A., Ballarin, E., La didattica delle lingue nel nuovo milennio. Le sfide dell'internazionalizzazione, Edizioni Ca'Foscari, 2018, pp. 135-148.