Spelling Correction for Estonian Learner Language

Anonymous Author
Affiliation / Address line 1
Affiliation / Address line 2
Affiliation / Address line 3
email@domain

Anonymouser Author
Affiliation / Address line 1
Affiliation / Address line 2
Affiliation / Address line 3
email@domain

Abstract

Spelling errors of second and foreign language (L2) learners are often specific compared to those made by native speakers. Language-independent spell-checking algorithms that rely on n-gram language models could offer a simple solution for improving learner error detection and correction due to context-sensitivity. The only open-source spell-checker available for Estonian is rule-based, so the aim was to evaluate the performance of bi- and trigram-based statistical spelling correctors on an error-tagged set of A2–C1-level texts written by L2 learners of Estonian. The newly trained spell-checking models were compared to existing spelling correction tools (open-source and commercial).

1 Introduction

It has been proposed that tailor-made spelling error correction systems are best suited for language learning purposes because the spell-checking tools developed for proficient users are often not able to correct specific mistakes, like real-word errors, i.e., errors that result in a valid homonymous word; diacritic errors, and pronunciation-induced errors. Whereas it is costly to develop rule-based error correction systems with learner-oriented explanations or neural spell-checking systems that require large quantities of training data, language-independent statistical spelling correction algorithms that use n-gram language models to analyse context could be a simple starting point for improving error detection and correction of L2 learner writings.

Currently, the only open-source spelling corrector developed for Estonian language is Vabamorf[1]. It is a rule-based library created by Filosoft Ltd. at the beginning of the 1990-s. It means that Vabamorf does not rely on context for candidate re-ranking.

First, we used Peter Norvig's (2007) approach that generates all possible spelling corrections by different edits, such as character deletions, insertions, replacements, and transpositions. The procedure is repeated to get correction candidates with two edits. The probability of candidates is estimated based on a unigram language model derived from a language corpus. We used a bigram language model in addition to a unigram model to add context-sensitivity.

Second, we applied the compound aware version of Symmetric Delete Spelling Correction (Symspell)[2]. The algorithm searches for candidate corrections with an edit distance of 1 or 2 based on deletions only, increasing the speed of spelling correction. A corpus-based bigram dictionary can be used, however, bigrams are only considered in ranking suggestions if no suggestions with an edit distance of 1 are found for a single token. Real-word spelling errors are currently not corrected by Symspell (Garbe, 2017).

Third, we tested Jamspell[3] that also uses a trigram language model for selecting the highest-scored correction candidate. Jamspell is based on a modified Symspell algorithm, optimized for speed and memory usage. The spell-checking library can process up to 5,000 words per second.

We compared the algorithms with three existing spell-checking tools: Vabamorf, and the commercial spellers offered by MS Word (Microsoft 365) and the Google Docs application.

2 Test Data and Evaluation

The correction output was evaluated on a set of 84 error-annotated proficiency examination writings

[1] https://github.com/Filosoft/vabamorf

[2] https://github.com/wolfgarbe/SymSpell

[3] https://github.com/bakwc/JamSpell

taken from the Estonian Interlanguage Corpus[4]. Divided between four proficiency levels (A2, B1, B2, and C1), the texts contain 9,186 words (excluding anonymized identifiers) and 309 spelling errors in total. We distinguished simple spelling errors and mixed errors, i.e., spelling mistakes co-occurring with another error such as word choice, inflectional form, or capitalization error. The error distribution is given in table 1.

Proficiency level	**Words**	**Simple spelling errors**	**Mixed errors**
A2	1,852	73	28
B1	2,186	71	12
B2	2,074	51	3
C1	3,074	68	3
Total	9,186	263	46

Table 1: Spelling correction test data.

The texts have been automatically annotated in the CoNLL-U format[5], using the Stanza toolkit[6], and manually error-tagged, indicating the error type, scope, and correction in the field for miscellaneous token attributes. While the custom tagset denotes various orthographic and grammatical errors, we only rated the detection and correction of words annotated to have a spelling error. Each text has been reviewed by two annotators, consulting a third Estonian language expert in case of disagreement. The annotation format allows for several corrections per token but is limited to one error annotation per sentence. This, however, has no significant effect on the analysis of spelling errors, which occur regardless of the sentence structure.

Spell-checking performance evaluation was mostly based on four metrics: **recall** – the percent of spelling errors detected; **precision** – the percent of correct changes made; **F0.5 score** – a combined measure of precision and recall that gives precision twice as much weight as recall, preferred to the harmonic mean (F1 score) due to the assumption that an error correction system's reliability is rather reduced by false and needless corrections than unproposed corrections (see Ng et al., 2014); the percent of **spelling errors corrected**.

[4] https://evkk.tlu.ee/about/us/

[5] https://universaldependencies.org/format.html/

[6] https://stanfordnlp.github.io/stanza/

We verticalized the system output and automatically compared it to the test set to detect changes and corrections matching the gold-standard annotation. Since L2 learners may not select the correct option from a list of suggestions (e.g., Heift & Rimrott, 2008) and such selection cannot be implemented in an automated workflow, e.g., using spell-checking as a GEC pre-processing step, we prioritized the speller's accuracy of defining the best correction. Thus, we focused on the highest-ranked suggestion. The cases of mixed errors were reviewed manually to find partial corrections fixing the spelling of an otherwise erroneous word. Both full and partial word corrections were considered in calculating the evaluation metrics.

3 Comparison of Spell-Checking Tools

The training material for building new statistical spell-checking models came from the Estonian National Corpus (ENC) 2019, which includes web corpora downloaded from Estonian websites as well as the Estonian Reference Corpus, Wikipedia corpora and the corpus of Estonian Open Access Journals (DOAJ) (Koppel & Kallas, 2020). Jamspell and Norvig's spelling corrector were trained on a random sample of 6 million sentences and over 82 million words retrieved from the Reference Corpus that represents the "standard" varieties of Estonian – mostly newspaper texts but also fiction, science and legislation texts from 1990–2008. The sample constitutes nearly half of the reference corpus; increasing the volume of the training set did not improve the correction results. Symspell, on the other hand, reached the best results with a uni- and bigram frequency dictionary based on the full ENC 2019 containing over 1.5 billion words. Even then, it performed very poorly compared to other tools. The comparison is summarized in tables 2 and 3.

Jamspell and Norvig's speller outperformed Vabamorf and MS Word speller in precision and F0.5 score, whereas the latter two had a higher recall. Google's spell-checker achieved a better performance, although Jamspell's precision, recall and F0.5 score came close to it. Despite a larger number of accurate corrections, Google also made more than twice as many unnecessary edits leading to broken words. All in all, Google corrected the highest proportion of spelling errors, while Jamspell also scored significantly better than the rest of the tools in terms of errors corrected. Sym-

Spell checker	Errors detected	Full corrections	Partial corrections	Broken words
Jamspell	207	129	29	24
Norvig	194	108	25	36
Symspell	119	45	9	19
Google	215	163	26	58
MS Word	215	108	24	30
Vabamorf	214	88	20	26

Table 2: Changes made by spell checkers.

Spell checker	F0.5	Precision	Recall	Errors corrected
Jamspell	68.1	68.4	67.0	51.1
Norvig	58.8	57.8	62.8	43.0
Symspell	39.0	39.1	38.5	17.5
Google	69.3	69.2	69.6	61.2
MS Word	56.4	53.9	69.6	42.7
Vabamorf	48.4	45.0	69.3	35.0

Table 3: Evaluation metrics of spell checkers (%).

spell broke the smallest number of words at the cost of a very low recall. The lowest percent of unnecessary edits among all changes was achieved by Jamspell and Vabamorf – 10.4% and 10.8% respectively. At the same time, 21.2% of words edited by Google did not need to be corrected.

If correct candidate suggestions were considered, the spell-checking tools would reach a higher precision, except for Google's speller that proposed only a single correction. Vabamorf's precision (72.5%) would increase the most, Jamspell's precision (72.3%) the least. It means that Jamspell is more likely to suggest an accurate correction with the highest confidence.

Compared to their open-source counterpart Vabamorf, both Jamspell and Norvig's speller benefit from relying on the context of erroneous words. For example, Vabamorf corrected the verbs **tõdida*∼*tõdeda* 'admit-INF' and **ludeda*∼*lugeda* 'read-INF' to *tüdida* 'get.bored-INF' and *kudeda* 'spawn-INF'. Interestingly, the rule-based spell-checker tended to replace other parts-of-speech with nouns, e.g., the adverb **lahtii*∼*lahti* 'open' was changed to *Lahti*, a location in Finland, and the adverb **nanuke*∼*natuke* 'a bit' to *januke* 'thirst-DIM'. Real-word spelling errors inducing homonymy were best handled by Jamspell that was able to make corrections such as **vaga*∼*väga* 'very' (*vaga* could be an adjective meaning 'pious, godly'); **töökohtu*∼*töökohti* 'job-PART.PL' (*töökohtu* could mean 'labour.court-GEN.PL'); and **kuued*∼*kuud* 'month-PART.SG' (*kuued* could be a numeral meaning 'six-NOM.PL' or a noun meaning 'coat-NOM.PL').

Like Google's spell-checker, Jamspell and Norvig's speller occasionally attempted to correct word choice and inflectional form, although merely 2–3 mixed errors were fully corrected. Otherwise, we only took these edits into account in the evaluation if they were unnecessary and resulted in a broken word. It can, however, be noted that Jamspell was more probable to make accurate lexical and grammatical corrections than Norvig's corrector, given a small edit distance, e.g., **ennem* 'rather'∼*enne* 'before', **kümne* 'ten-GEN.SG'∼*kümme* 'ten.NOM.SG'. Similar to Vabamorf and MS Word, Norvig's speller replaced some proper nouns with common nouns, e.g., *Kemeris* 'Kemer-IN.SG' referring to a Turkish location was corrected to *Keeris* 'vortex.NOM.SG'. Such behaviour was the most characteristic to Vabamorf which also proposed changes to rather common first names, e.g., *Nadja*∼*Andja* 'giver'. Furthermore, some unnecessary edits made by Google, MS Word and Symspell were caused by splitting compound words.

In conclusion, two of the tested statistical spell-checkers achieved a better precision in correcting Estonian L2 learners' spelling errors compared to the existing open-source speller Vabamorf and MS Word speller. Jamspell's performance was similar to Google's spelling corrector, the main difference being that Google corrected more spelling errors at the cost of making more unnecessary edits. Therefore, Jamspell should be favoured if the priority is to minimize needless corrections.

4 Jamspell Correction Models

We experimented with different training data to see if we can improve Jamspell's efficiency in learner spelling error detection and correction. The training sets are listed in table 4. On the one hand, we combined the Estonian Reference Corpus with the DOAJ and Wikipedia corpora of ENC 2019. These subcorpora contain, to a large extent, language-edited texts. As the Reference Corpus constitutes the majority of this training set, we also extracted a more balanced sample, which includes an equal amount of randomly chosen sentences from the Reference Corpus and Wikipedia corpora as well as the whole DOAJ corpus (442,663 sentences). On the other hand, we trained Jamspell on the Estonian Web corpus 2019 that comprises a more diverse selection of texts, from informal blog posts and forum discussions to periodicals and educational materials. We used the full corpus and a sample similar in size with the Reference Corpus sample. Thirdly, we merged the Reference Corpus and Web 2019 material in an equal ratio and in a ratio of 10:1, giving emphasis to the more "standardized" texts and using the web texts to add variation to the dataset.

The results are presented in tables 5 and 6. In case of similar training datasets (full corpus and sample), the lower-performing correction model has been omitted. Models trained on a sample of the Reference Corpus and its combination with other edited subcorpora achieved better or similar results compared to the models trained on full text sets. Contrary to that, the model trained on the whole Estonian Web 2019 performed better than the model based on the web sample in all aspects.

The comparison of the Jamspell models reflects the well-known trade-off between precision and recall. The highest precision and F0.5 score were achieved by the model trained on Estonian Web 2019. It was the least likely to make unnecessary corrections but also to detect words with a spelling error. At the same time, the initial model trained on a Reference Corpus sample scored highest in recall and the amount of errors corrected. Expectedly, using the 1:1 Reference + Web sample yielded results closer to the web corpus model, while the performance obtained with the 10:1 Reference + Web sample was more similar to the Reference Corpus trained model, and the model trained on the Reference + Wikipedia + DOAJ sample, which had almost identical scores with slightly fewer corrections and unnecessary edits.

Training corpus	Sentences	Words
Web 2019	40,880,346	512,567,596
Reference + Wikipedia + DOAJ	16,935,524	230,066,343
Reference	13,173,122	180,944,778
Web 2019 sample	6,000,000	75,237,791
Reference sample	6,000,000	82,401,187
Reference + Web 1:1	6,000,000	78,855,570
Reference + Web 10:1	6,600,000	89,921,477
Reference + Wikipedia + DOAJ sample	4,172,777	55,743,160

Table 4: Data for training Jamspell models.

Concerning the relation between the training corpus type and size, and the performance of the spell-checking model, we may infer that a smaller, "standard language" dataset rather facilitates higher recall. Increasing the dataset introduces more noise, thus the errors are outlined less clearly. A much larger and more diverse language model leads to higher precision; decreasing the dataset reduces lexical variation and entails more unnecessary edits. For comparison, the Web 2019 trigram model consists of 279.1 million trigrams, whereas the model trained on the Reference Corpus sample has 52.8 million trigrams. The choice of the most suitable model depends on the purpose – whether we want to maximize the amount of errors detected and corrected, minimize the amount of needless corrections, or find a middle ground.

Training corpus	Errors detected	Full corrections	Partial corrections	Broken words
Reference sample	**207**	**129**	**29**	24
Reference + Wikipedia + DOAJ sample	186	122	25	26
Reference + Web 10:1	186	116	26	18
Reference + Web 1:1	172	113	23	20
Web 2019	166	115	16	**10**

Table 5: Jamspell models ranked by the number of spelling corrections.

Training corpus	F0.5	Precision	Recall	Errors corrected
Web 2019	**69.1**	**74.4**	53.7	42.4
Reference sample	68.1	68.4	**67.0**	**51.1**
Reference + Web 10:1	67.5	69.6	60.2	46.0
Reference + Wikipedia + DOAJ sample	67.3	69.3	60.2	47.6
Reference + Web 1:1	67.2	70.8	55.7	44.0

Table 6: Evaluation metrics of Jamspell models (%), ranked by the F0.5 score.

In summary, the results are promising compared to the precision and recall of learner spelling error correction accomplished in other languages (e.g., Bexte et al., 2022; Kantor et al., 2019).

5 Conclusion and Perspectives

This study has demonstrated the benefit of statistical context-sensitive spelling correction for processing L2 learner writings. Jamspell that uses trigram contexts of words for spell-checking could correct real-word errors and other learner-specific spelling errors more efficiently than other tested open-source spellers. It also outperformed MS Word speller, achieving precision and recall comparable to Google's contextual spell-checker. The evaluation of different Jamspell correction models revealed that using a web corpus as training material increases precision, while using a reference corpus increases recall.

We consider the current correction models a decent baseline for further development. Their performance could be improved, e.g., by employing learner spelling error frequency data and named-entity recognition to avoid false name corrections. We acknowledge that the results might have been different if we had implemented Norvig's spell-checking algorithm on a trigram language model. The tested spell-checking tools and models should also be evaluated on a larger error-annotated set of writings by L2 learners as well as native speakers. Such a gold-standard dataset of appr. 8,000 sentences is in development for Estonian.