Man or machine: Evaluating Spelling Error Detection in Danish Newspaper Corpora

Eckhard Bick[1], Jonas Nygaard Blom[1], Marianne Rathje[2], Jørgen Schack[2]
[1]University of Southern Denmark, eckhard.bick@gmail.com, blom@journalism.sdu.dk
[2]The Danish Language Council, {mr, schack}@dsn.dk

Abstract

This paper evaluates frequency and detection performance for both spelling and grammatical errors in a corpus of published Danish newspaper texts, comparing the results of three human proofreaders with those of an automatic system, DanProof. Adopting the error categorization scheme of the latter, we look at the accuracy of individual error types and their relative distribution over time, as well as the adequacy of suggested corrections. Finally, we discuss so-called artefact errors introduced by corpus processing, and the potential of DanProof as a corpus cleaning tool for identifying and correcting format conversion, OCR or other compilation errors. In the evaluation, with balanced F1-scores of 77.6 and 67.6 for 1999 texts and 2019 texts, respectively, DanProof achieved a higher recall and accuracy than the individual human annotators, and contributed the largest share of errors not detected by others (16.4% for 1999 and 23.6% for 2019). However, the human annotators had a significantly higher precision. Not counting artifacts, the overall error frequency in the corpus was low (~ 0.5%), and less than half in the newer texts compared to the older ones, a change that mostly concerned orthographical errors, with a correspondingly higher relative share of grammatical errors.

Keywords: Spell- and grammar checking, Danish Newspaper corpora, Spelling quality evaluation

1. Introduction

Today, spell- and grammar checkers are widely used to assist human proofreading. For many text types, human proofreading is reduced to accepting, discarding, choosing from or editing spellchecker suggestions, in a kind of post-editing workflow. But which is more effective, human proofreading or automatic spellchecking? What are the two methods' error detection rates? Are there certain kinds of errors that can be more reliably handled by spellcheckers than others?

In this paper, we will address these questions for the professional, and as such high-quality, genre of printed newspapers, i.e. using data that has, most likely, *already* undergone either spellchecking or proofreading or both. We will show, for Danish data, that even in this low-error scenario, for each additional human proofreader, or by running a new kind of spellchecker, additional errors can be found. That combining human and automatic spellchecking is necessary for maximizing error detection is also supported by English results. For instance, Tetreault et al. (2017), in their study on grammatical errors and fluency, found that humans outperformed automatic systems on this task, but also that individual humans had an edit-distance score of only 63.2.

Our second focus is the evaluation of a specific spell- and grammar checker, DanProof (Bick, 2015), and its performance in the newspaper domain. As pointed out by Sahu et al. (2020), in spite of the ubiquity of the tools as such, there are relatively few studies that evaluate proofing tools, and to the best of our knowledge, DanProof is the only Danish system that has been systematically evaluated.[1]

2. Project Background and Data

The work presented here focuses on Danish and was carried out in connection with a diachronic study on the prevalence of spelling errors in Danish newspapers, the original research question being whether the number of spelling errors today was higher or lower than twenty years ago, and what kind of errors were most common now and then. The study was motivated by a widely held folk perception[2] of a deterioration of spelling proficiency in newspapers, but was able to refute this claim (Rathje et al., 2023), settling inconclusive or contradictory findings from earlier studies, e.g. by Kristensen et al. (2007), who claimed a deterioration, and Diderichsen and Schack (2015), who found an improvement for at least the category of "non-words". This also hints at a possible difference between Danish and English, for which Beede and Mulnix' (2017) have claimed that spelling error rates persist in digital news at a level comparable to pre-digital data. One possible explanation could be that Danish, as a less-resourced language, has only recently profited from an improvement in the quality of automatic spellchecking that had been factored in for English long ago.

For our new Danish study, two newspaper corpora of comparable size and composition were compiled, for 1999 and 2019, with ca. 100,000 words each, from the same seven mainstream (printed) newspapers.[3] Representativeness was ensured by sampling

[1] (Bick, 2015) also offered evaluation results for DanProof, but for a different target domain. In section 6, we will make a comparison between the two studies.

[2] Rathje et al. (2023) found that 86% of respondents in their Facebook inquiry thought that newspapers "had more errors today".

[3] Archival text data was provided by *Infomedia A/S.* The seven newspapers were *B.T., Berlingske, Ekstra Bladet, Information, Jyllands-Posten, Politiken* and *Weekendavisen.* The corpus was compiled such that their relative shares match the number of readers per newspaper, using data from Index Danmark/Gallup (https://webtest.kantargallup.dk/reports).

SIGUL2024 Workshop, pages 204–211

chunks of about 250 words from each article. All in all, 520 errors were found[4] in the 1999 data, and 230 errors for 2019 (cf. section 5.3), a marked difference corroborating Diderichsen and Schack's claim of impoved newspaper spelling standards.

Error annotation was independently performed by three human language professionals[5] and by the afore-mentioned automatic system, DanProof, a command-line version of the commercial interactive tool RetMig (https://retmig.dk). Each error candidate, flagged by man or machine, was then discussed in plenum and differences of opinion settled by resorting to the official Danish spelling dictionary, *Retskrivningsordbogen,* using the edition valid for the period in question, or by agreeing on a principled handling of problematic cases such as loan words, names and abbreviations.

3. Automatic Spell- and Grammar Checking: DanProof

The most basic spellcheckers employ a simple list-based methodology flagging words as errors if they are not on an approved fullform list, and suggesting similar words from the same list as corrections. Here, similarity is usually defined as editing distance[6] and often combined with frequency ranking (e.g. Singh et al., 2016). To improve coverage, especially for morphologically rich languages, productive inflection, affixation and compounding may be provided for through some kind of morphological analysis (e.g. *Hunspell*[7]). This method is not, however, sufficient for handling real word errors and grammatical errors, or for adequately ranking correction suggestions. More advanced tools therefore make use of contextual and lexical knowledge, either through contextual and grammatical rules, or through machine learning. Today, the latter is more common than the former, employing various strategies for different aspects of a spellchecking pipeline. For instance, De Amorim and Zampieri (2013) suggest unsupervised word clustering as an alternative to the aforementioned editing distances for establishing word similarity, while Choe et al. (2019) use sequential transfer learning for building an educational grammar correction system. Machine learning can also be used to combine spellchecking with other tasks, as shown by Gosh and Kristensen (2017), where neural networks are employed to integrate text correction with text completion, achieving 90% word accuracy for a Twitter typo dataset.

DanProof itself is a rule-based system targeting both orthographical and grammatical errors at the same time.

Figure 1 illustrates the architecture of the DanProof program pipeline.

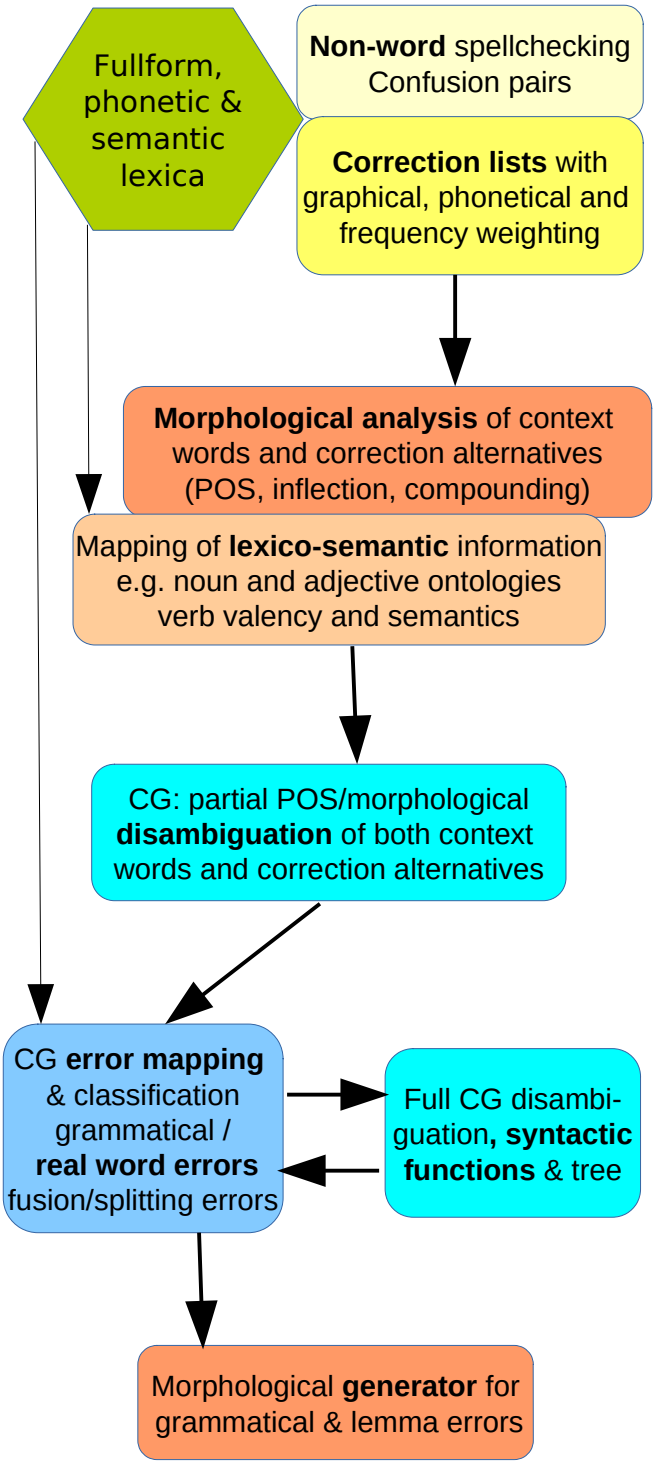

Figure 1: System flow chart (DanProof)

In line with the rule-based approach, there is a special focus on explicability and pedagogical

[4] These are the aggregate numbers for the three human proof readers, plus the automatic system.
[5] Two of these were employees of the Danish Language Council, the institution in charge of the official Danish spelling rules and dictionary, the third was a university researcher.
[6] Editing distance (or Levenshtein distance) means the minimum number of letter insertions, deletions or substitutions needed to transform one wordform into another.
[7] https://hunspell.github.io/

aspects, as all errors are classified and, if desired,[8] explained and backed up with a morphosyntactic analysis. Also, emphasizing contextual ranking of correction suggestions benefits both user-friendliness (in an interactive setting) and stand-alone error annotation, e.g. unsupervised corpus cleaning. In this set-up, the first module[9] flags non-words, as well as some commonly confused real words, and suggests spelling corrections with both an overall weighting and separate numerical weights based on graphical[10] and phonetic[11] similarity as well as corpus frequency. After adding morphological analyses for both real words and correction suggestions, morphosyntactic Constraint Grammar[12] (CG) disambiguation rules then weed out replacement wordforms that clash with Danish language rules, in parallel with ordinary POS and inflectional disambiguation, and while building a syntactic parse tree. A second spellchecking module addresses remaining ambiguity and real word errors, not least grammatical errors, using dedicated error mapping and disambiguation rules targeting (and at the same time naming) individual error types. This module is run twice, at different points in the program pipe – first early on, before complete morphological disambiguation, to prevent for instance agreement errors from triggering incorrect POS disambiguation, then a second time after full disambiguation and with contextual knowledge of the syntactic tree. Semantic information, such as ontologies for nouns and adjectives and framenet categories for verbs (Bick, 2011) are added with a lexical mapper early on and available, albeit with limited disambiguation, throughout the whole program pipe.

4. Error Types

Before error classification proper, error candidates were discarded if they were either deemed as "out-of-scope" or "corpus artifacts". Out-of-scope errors would be, for instance, intentional errors (e.g. the use of 'z' instead of 's', as an onomatopoeic marker, in *'renzezkum'* [cleaning foam]), misspellings of out-of-vocabulary (OOV) names (e.g. *Michoacan* vs. *Michoacán*) or widely used upper-casing of non-dot abbreviations such as *TV* or *CD,* which the official spelling norm in 1999 would have in lower case.[13] Corpus artifacts are errors caused by encoding or format conversion (e.g. loss or insertion of spaces, hyphens and accents) and will be treated in detail in the evaluation section.

The remaining, "true" errors were originally classified using a typology introduced by Jørgen Schack (Rathje et al., 2023) and based on the spelling rule section of the official Danish spelling dictionary *(Retskrivningsordbogen).* For the sake of error detection evaluation, to ensure compatibility with the automatic system, we will here use a slightly different category set based on DanProof's own error tagging. In this scheme, the following error categories can be distinguished:

1.) **core-orthographical**, non-grammatical spelling errors with one or more wrong or wrongly placed letters, not involving casing or non-letter characters, e.g. *verjtrækning* for *vejrtrækning* (breathing).

2.) **splitting errors**, typically compounds (e.g. *cykel[ ]kurven* [bicycle basket]), prefixes (e.g. *super[ ]sexet* [very sexy]) or 2-part adverbs (*langt fra* [far away from] for *langtfra* [not at all]

3.) **fusion errors**, e.g. *henover* for *hen over (*across) or *caffelatte* for *caffe latte* or *engang* (once=then) for *en gang* (once=not twice)

4.) **hyphenation errors**, i.e. missing or spurious hyphens, e.g. *ånds-revolution* (correct: *åndsrevolution* [spiritual revolution]) or *15 års fødselsdag* (correct: *15-års fødselsdag* or *15-årsfødselsdag* [15-year birthday]). Possibly inspired by English usage, hyphens are often omitted after attributive proper nouns, e.g. *Wampanoag høvdingen* (correct: *Wampanoag-høvdingen* [the Wanpanoag chief])

5.) **apostrophe errors**, where an apostrophe is missing, typically before the genitive-s after upper case abbreviations or numerical roots, e.g. *IBMs* (correct: *IBM's*), *60erne* (correct: *60'ernae* [the 1960s]), or – sometimes – wrongly inserted, e.g. *logo'er* (correct: *logoer* [logos]).

6.) **casing errors**, i.e. confusion of upper case and lower case, for instance after a colon or in complex proper nouns (e.g. *von humboldt* for *von Humboldt*).

7.) **word-level errors,** defined as missing, spurious or wrong words. While spurious words are often repetitions and as such easy to detect, e.g. *en af en de mest …*[one of one the most …] (correct: *en af de mest …* [one of the most …]), insertions are often syntactically and replacements semantically motivated, representing progressively more difficult tasks for an automatic system.

[8] This is the case for *Retmig,* the interactive version of *DanProof,* which can be used on-line in a browser, or with *Word, Libre Office, Google Docs* etc.

[9] The basic method goes back to a precursor tool, *OrdRet,* and is described in detail in (Bick, 2006).

[10] DanProof's graphical similarity metric goes beyond edit distances (number of letter substitutions, insertions or deletions needed to correct a word) by also integrating keyboard distances and letter adjacency likelihoods.

[11] Phonetic similarity between error word and correction suggestion is particularly relevant for children and language learners, as pointed out by Downs et al. (2020) in their evaluation of *KidSpell*, and helps ranking multiple correction options.

[12] Constraint Grammar (e.g. Bick, 2023) is a context-based method for automatic morphosyntactic, structural and semantic annotation and disambiguation.

[13] More specifically, the latter were ignored, because they were out-of-scope for *DanProof,* which only knows the current spelling norm for abbreviations and does not have a historical "1999 mode".

8.) **grammatical errors** or morphological errors are existing word forms that are wrongly inflected given the sentence context. DanProof employs various subcategory tags comprising not least agreement errors concerning definiteness (@def/@idf, e.g. *en gigantiske fortrop* [a huge vanguard], number (@sg/@pl, e.g. *sin[e] forældre* [one's parents]), gender (@utr/@neu, e.g. *et sådan[t] system* [such a system]) or finity (@inf/@vfin/@impf/@pcp, e.g. *at kommer til* [to arrive at]). Notorious are the so-called 'r-errors' (missing or spurious r-endingsc[14]), which are considered uneducated in Danish and caused by the silent '-r' ending marking the present tense and the plural of nouns. Finally, the category includes adverbial '-t' errors (@adv-t), especially where adverbs are formed from adjectives by means of inflection, e.g. *offentlig[t] ejet* (publicly owned).

In terms of error detection, an important distinction has to be made between non-word errors (which are always wrong) and real-word errors (where the wordform as such does exist). This distinction is in principle orthogonal to the above error categorization, but some correlation is to be expected. Thus, non-word errors are typical of category (1), while grammatical errors (8) and word level errors (7) are always real-word errors. Accidental splitting (2) and fusion (3) will mostly result in non-words, while the more common compound splitting and some ambiguous fusion of function words may result in real-word errors.

As real-word errors are only wrong in context, an automatic spellchecker needs to "understand" this context linguistically, either in a rule-base fashion or implicitly through machine-learned pattern recognition. Non-words, on the other hand, are in principle easy to detect automatically given an unabridged list of correct word forms. The human brain, however, is trained to recognize known patterns, and annotators may sometimes overlook this kind of error, if only a single letter is affected, for instance in consonant clusters. In terms of automatic error annotation, non-words are harder to be sure of for Danish than for English, because word list coverage is affected by the fact that Danish has a lot of productive compounding and loan words.

DanProof addresses this problem by trying to annotate non-listed, but "good" words as @new rather than wrong, drawing on compound analysis and letter patterns of loan words. In addition, non-words that do not have a close graphical or phonetical correction suggestion, are marked as dubious (@check!). Finally, named entity recognition (NER) is used to flag unknown names as *not* wrong, tagged @proper. By filtering out @new and @proper tags, or even the less safe @check! tags, a large amount of false positives can be avoided, and precision improved compared to other spellcheckers that do not recognize OOV compounds and names as such.

[14] In Danish, an r-ending is used to distinguish finite verbs from infinitives, and also as a plural marker for nouns.

5. Evaluation

5.1 Scope and Data

In this section we perform a comparative evaluation of human and automatic error detection (5.2) and provide a break-down of different error types with respect to frequency (5.3). Furthermore, the performance of DanProof is evaluated in terms of detection recall, precision and F-score[15] (5.4), as well as correction adequacy (5.5), discussing strengths and weaknesses. The evaluation gold standard was arrived at by aggregating the markings of all annotators, as well as the automatic system, resolving inter-annotator differences through discussion and by consulting the official spelling dictionary and rules. Both news corpora (i.e. covering the years 1999 and 2019, cf. section 2) are used for the evaluation, amounting to about 200,000 words in all. Given the equal size and composition of the two corpora, we make diachronic comparisons between 1999 and 2019 where relevant. Finally, the prevalence and handling of corpus artefact errors is discussed (5.6), evaluating DanProof's use as a corpus cleaning tool.

5.2 Error Detection Performance

Tables 1 and 2 present the error detection recall, precision and F1-Score for the individual annotator, as well as the contribution of "exclusive" errors, found only by one annotator (last column).[16]

	Recall	Precision	F1-score	errors found only by
Human A	46.4	94.8	62.3	3.5 %
Human B	48.0	88.2	62.2	8.8 %
Human C	57.1	97.0	71.9	3.1 %
System	73.3	82.5	77.6	16.4 %

Table 1: Error detection performance, 1999 data

	Recall	Precision	F1-score	errors found only by
Human A	41.2	98.0	58.0	5.6 %
Human B	35.6	91.2	51.2	10.3 %
Human C	43.8	100	60.9	5.2 %
System	71.7	64.0	67.6	23.6 %

Table 2: Error detection performance, 2019 data

As can be seen, there was considerable variation in F-scores for error detection (51.2 to 77.6), with

[15] Recall is calculated as R=c/(c+fn), precision is calculated as P=c/(c+fp) and the F-score accuracy as Fß=(1+ß)*R*P/(R+P*ß), with c=correctly identified errors, fn=false negatives (errors missed), fp=false positives (non-errors mistaken for errors), and ß a weighting coefficient, set to 1 for balanced weighting of recall and precision.

[16] Here, a high recall means being good at finding errors, while a low precision means marking errors that were not actually errors. However, scoring low at either does not necessarily preclude finding errors that others did not find (Human B), suggesting a certain variation as to which error types people are good at.

DanProof outperforming human annotators in terms of F-score for both corpora. A closer look at the underlying recall and precision figures, however, shows a marked difference between humans and the automatic system in that the latter excelled in recall, while humans had much better precision. In other words, a human annotator might overlook an error (or not be sure of officially sanctioned spelling variants), but would have a much better intuition about acceptability if confronted with out-of-lexicon items such as new loan words, brands and word games. This difference could be made explicit by using F scores with ß<1,[17] which would weight precision higher than recall. But ultimately, such considerations are task-dependent, and for *finding* as many errors as possible (as was the case in the newspaper spelling study), recall is more important, as false positive markings can be weeded out in a discussion phase, while (overlooked) false negatives will obviously not be recoverable by a discussion phase.

Interestingly, the combined number of errors identified was much larger than the individual annotator's contribution. Thus, errors found by only one annotator or only by DanProof added up to 31.8% for the 1999 corpus and 45.3% in 2019, with DanProof making the largest contribution, with 16.4% in 1999 and 23.6% "exclusive" error findings in 2019. Conversely, only 18% (1999) resp. 15% (2019) of errors were marked by all human annotators, or 15.8% resp. 13.7% by both all humans and the automatic systems.

5.3 Error Frequency

As would be expected for redacted and published material, spelling errors were relatively rare in both newspaper corpora, with a frequency of 0.52% of words in the older and 0.23% in the newer data.[18] The fact that there were about half as many errors in 2019 compared with the 1999 data probably marks a clear tendency even without intermediate data points, given that spelling proficiency is not a chaotic system in mathematical terms and likely to follow a monotonous curve, due to factors like spelling reforms, school and journalist education and the use, ease and quality of automatic spellcheckers. Table 3 provides a comparative break-down of error types for the two corpora.

Error type	‰ 1999	‰ 2019	share of 1999	share of 2019
letter sequence (spelling)	1.63	0.73	31.8	31.3
grammatical (morphology)	0.54	0.44	10.5	**18.9**
word-level (missing, extra, wrong)	0.26	0.23	5.1	**9.9**
splitting error	0.44	0.14	8.6	6.0
fusion error	0.45	0.09	**8.8**	3.8
hyphenation	0.72	0.16	**14**	6.9
apostrophe	0.55	0.20	10.7	8.6
casing (upper/lower)	0.54	0.34	10.5	14.6

Table 3: absolute & relative frequency of error types

As can be seen, the overall tendency of lower error rates in the newer data is, by and large, confirmed also at the level of individual error categories. However, the change is not uniform, and in relative terms, grammatical errors (covering inflection and agreement, in particular) and word-level errors appear to be on the rise. One possible explanation is that this type of error is always a real-word error, i.e. impossible to spot with ordinary, list-based spellchecking. And as list-based spellcheckers have become better and more commonly used, the proportion between surviving error types may well have changed in favor of real word errors (bold face, 2019).

Conversely, there were more fusion and hyphenation errors in 1999 (bold face). Many of the former were caused by a distinction between adverbial (fused) and prepositional (split) use of expressions like 'overfor'/'over for' (opposite ADV, opposite of PRP) – a distinction that for many cases has been dropped in the current Danish spelling rules. The 1999 hyphenation errors were mostly spaces instead of hyphens, possibly because older spellcheckers would not recognize the hyphenated form, but accept the two parts on their own when split.

5.4 Error Types: Easy or Difficult?

Table 4 illustrates the performance of the automatic system by error type, for both corpora. Here, it is important to look at recall and precision rather than just F-scores. High recall and low precision means that a given error type is well-covered, but comes at a high price in terms of false positives. Low recall and high precision means that most error flaggings are sound, but at the price of overlooking many false negatives.

Error type	R 1999	P 1999	F 1999	R 2019	P 2019	F 2019
letter sequence	84.7	80.2	82.4	72.6	**45.7**	56.1
grammatical (morphology)	77.8	71.2	74.4	79.5	77.8	78.6
word-level	**46.2**	80.0	58.6	**34.8**	100	51.6
splitting error	65.9	80.6	72.5	71.4	90.1	79.7
fusion error	77.8	100	87.5	77.8	100	87.5
hyphenation	69-4	78.1	73.5	62.5	71.4	66.6
apostrophe	**85.5**	100	92.2	**95.0**	100	97.4
casing (upper/lower)	42.6[19]	76.7	54.8	73.5	61.0	66.7
all	73.3	82.5	77.6	71.7	64.0	67.6

Table 4: DanProof performance by error type

[17] With a strong precision weighting, at ß=0.5, DanProof ranks 2nd for 1999, but lower than all human annotators for 2019. With a more moderate ß=0.8, however, DanProof still leads for both corpora, even with precision weighted more than recall.

[18] Rathje et al. (2023) report a slightly higher frequency of 0.55% and 0.24%, respectively, caused by different leniency for the category of out-of-scope errors.

[19] The low recall for this category is an outlier, where almost half of all cases were caused by lower-casing of only two items, 'EU-**p**arlamentet' and '**d**ankort'.

We see a balanced performance (without big differences between R and P) for hyphenation and apostrophe errors, the latter also having the highest F-score in both corpora. For categories affecting word number, however, i.e. splitting, fusion and word-level errors, precision clearly outperforms recall, meaning that once an error is spotted, it is fairly safe (i.e. few false positives), but that the error patterns are difficult to see for the machine. This is especially true of word-level errors of the type "missing word" and "wrong word", which usually ask for a deep understanding of the sentence or knowledge of fine-grained language usage nuances.

For one category, letter sequence errors, there is a marked, and at first glance inexplicable, performance deterioration between 1999 and 2019. However, this should be seen on the background of a much lower absolute error frequency (1 letter sequence error per 1,500 words), with many easy errors gone due to increased and better spellchecking at production time. In other words, the remaining spelling errors are likely to be harder, and[20] detecting them comes at a higher price in terms of false positives (lower precision). Another explanation could be that DanProof's lexicon has a better list coverage for the older texts, as the system has been built over more than 15 years and depends on manual lexicon additions.[21]

It could be interesting to compare these results with those for other text types. Thus, the best system in an early French study on student essays (Starlander and Popescu-Belis, 2002) achieved lower scores for grammatical errors (F=58.4), but performed better than DanProof for letter/spelling errors (F=89.3). The latter seems to underscore our above hypothesis that a higher frequency of spelling errors correlates with better scores (student essays, and 1999 newspapers versus 2019), while a lower frequency may mean more difficult errors and increases the risk of false positives (newspapers, especially 2019).

5.5 Correction Adequacy

For the binary error types of splitting, fusion, missing or spurious hyphen, apostrophe and casing errors, spotting the error implies being able to provide an adequate correction, by simply toggling the orthography feature in question, yielding 100% suggestion adequacy. Given a full-fledged morphological generator, this is also true of most grammatical errors. For phonetic, typographical and other letter-based misspellings, however, this is not true. Here, it is one thing to spot an error, another to come up with an adequate correction. Unlike the interactive on-line edition (RetMig), our command-line version of DanProof provided exactly one correction (or none), not a ranked list. For the 1999 corpus, this suggestion was wrong in 16.7% of all correctly identified letter-errors, and missing in 4.3%, amounting to a correction adequacy of 79%. For 2019, the numbers were 7.5%, 1.9% and 90.6%, respectively. This corresponds to a combined, reduced detection+suggestion F-score for this error category of 71.2 for 1999 and 52.2 for 2019. Due to the 100% suggestion adequacy of most other error types, overall F-scores are less affected, with detection+suggestion F-scores of 73.8 and 66.3, respectively, for the two corpora. No comparable evaluation data could be found for other Danish spellcheckers, but the numbers compare favourably with the similar "E-measure"[22] used by Näther (2020) in his evaluation of English spellcheckers on artificially generated Wikipedia errors, where the best product (Grammarly) scored 46.98, and a neural net transformer trained on the same type of data scored 62.24. For French, Starlander and Popescu-Belis (2002) reported correct suggestions (though not necessarily top-ranking) for 73.9% of correctly flagged errors.

5.6 Corpus Artefacts

Not everything that looks like an orthographical error is human-made. Thus, different phases of corpus creation may introduce additional errors, one well-known example being OCR errors or pdf-to-text conversion errors. But even for corpora based on electronic text sources, as was the case for our newpaper data, errors may be introduced when converting from different native text processor formats to the encoding chosen for the corpus itself, or when producing the .txt format to be used for automatic analysis. Here, a common problem is artificial word fusion or splitting caused by e.g. turning soft hyphens into hard hyphens or by not turning various delimiter characters into spaces or newlines. Another problem is the conversion of accented or otherwise special characters. Also, conversion programs are often written without using linguistic resources and contextual rules, resulting in, for instance, artificial sentence splitting by mistaking abbreviation dots for fullstops.

A human annotator will recognize and ignore many of these errors, but for an automatic system the difference is not obvious, and the artefacts will be annotated just like other error. By changing the context (e.g. faulty sentence separation or mistaking fused words as OOV nouns), artefacts may even affect annotation performance for real errors. On the other hand, recognizing artefactual errors will allow a spellchecker to be used for automizing tedious tasks like corpus cleaning, format conversion checking and OCR postprocessing. Table 5 quantifies the performance of DanProof in this respect and provides a breakdown of error types for this task.

In absolute terms, artefact errors were a much bigger problem in the newer corpus. Thus, in 2019, there was 1 artefact error for every 2 real errors, while the proportion was 1 to 10 for the 1999 corpus. Also, for 1999, most artefacts were only marked by

[20] For a hypothetical, error-free newspaper, *all* error flags would be false positives, and precision zero.

[21] An objective indicator for this is the fact that the number of OOV words marked either @new or @check! was 28% higher in 2019 for the former and 68% higher for the latter.

[22] A detection+correction F-score average over all error types, more or less the same types as in our own study. The scheme included a NONE type for error-free input, with F=97-98 for the best systems, that – all other things equal – would have resulted in somewhat higher E-scores.

DanProof (90.4%), while its exclusive share was lower for 2019 (68.7%).

Artefact read as	1999 %	2019 %	2019 "R"
letter sequence (spelling)	17.6	3.6	100
grammatical (morphology)	-	-	-
word-level (missing, extra, wrong)	-	-	-
splitting error	31.4	65.1	**98.1**
fusion error	7.8	10.8	77.8
hyphenation	23.5	14.5	45.8
apostrophe	15.7	0.6	100
casing (upper/lower)	2.0	4.8	100
unrecognized (proper/new)	2.0	0.6	100
only marked by DanProof	90.4	68.7	

Table 5: Artefact errors

In relative terms, splitting errors were the largest category, especially for 2019 (bold face). In the latter, splittings mostly affected double-dot abreviations, with an internal space after the first dot (*'f. eks.'* [e.g.]). In 1999, there were spurious word-internal hyphens and spaces,[23] e.g. *med- redaktør* (correct: *medredaktør* [co-editor]), likely caused by line-break hyphenation. One reason for the larger prevalence of letter-spelling and apostrophe artefacts in 1999 was the rewriting of 'é' as '+e', and the replacement of apostrophs with spaces.

Since DanProof does not have a separate "artefact" tag along with the error category tag, false positives are indistinguishable from ordinary false positives, and calculating precision does not make sense. Recall can be calculated, but with the caveat that the human annotators did not always mark artefacts that did not look like a spelling error to them. Thus, only a few artefacts were marked by a human annotator in the first corpus (1999), and none without a DanProof mark at the same time. We therefore only provide recall figures for 2019. Here, hyphenation artefacts proved to be the most difficult category (R=45.8), followed by fusion errors (R=77.8). All other categories were reliably flagged.

6. Conclusion and Discussion

We have shown that the detection of spelling errors in high quality texts such as printed newspapers profits from a combination of multi-person human proof reading and automatic spellchecking. Thus, a single proof reader risks overlooking half of all errors (recall of 43-64 %), the problem being more pronounced if the texts contain fewer errors to begin with, making the 2019 corpus harder than the 1999 corpus, which had more than twice as many errors. Using multiple annotators helped,[24] but the largest contribution in terms of recall gain came from adding an automatic spellchecker, DanProof, with 23.6% exclusive error hits for 2019 and 16.4% for 1999. However, the spellchecker's high recall contribution came at a price in terms of false positives, with the human annotators, on average, flagging errors with a significantly[25] higher precision, especially in the low-error-rate-scenario (2019).

DanProof achieved satisfying F1-scores of 77.6 and 67.6 for the 1999 and 2019 data, respectively. However, performance was not uniform across error types. Thus, the system did best for apostrophs and worst for word-level errors, and it performed better for orthographical spelling errors than for grammatical errors, and better for fusion errors than for splitting errors and hyphen-errors. In a real-world scenario, aiming for a reasonable error reduction at low human post-editing cost, it would make sense to filter out DanProof suggestions for low-performance errors, and – in particular – low-precision errors, or to build an arbiter system with multiple spellcheckers providing confidence ratings based on the systems' recall and precision for different error types. Arguably, differences in method and system architecture could become an asset in such a set-up, and it would make sense to combine a rule-based system like DanProof with a spellchecker based on machine learning. Thus, for the category of compound splitting errors, neural networks achieved a higher recall than a competing CG system for Sámi (Wiechetek et al., 2021), with only a moderate fall in precision.

Though it seems safe to assume that automatic spellchecking was used in both 1999 and 2019, it is a limitation of our study that we cannot know for sure if and which spellcheckers were used by the individual newspapers. It is likely that our DanProof evaluation is "unfair" in the sense that it amounted to running the system as the last element in a chain of prior automatic spellchecking and human postediting, which probably affected both recall and precision percentages, as many "easy" errors had already been corrected at production time, aggravating the low-error-rate effects noted when comparing the 1999 corpus with the "cleaner" 2019 corpus. A case in point in this respect is our finding that the relative share of grammatical errors (and hence the difficult real-word errors) increased between 1999 and 2019, notwithstanding the overall lower error rate in the latter.

7. Ethical Considerations

As our corpora are based on published and printed material and only used internally, this work does not raise any ethical concerns regarding GDPR. The main software used, DanProof, is a rule-based system and as such saves the computing power needed for training and using large language

[23] With both hyphen and space, these were counted as splitting artifacts, without the space as a hyphenation artifact.

[24] Even in this multi-annotator setup, it is reasonable to assume that errors may have been overlooked. However, the "uniqueness share" (5% on average for the three humans, cf. table 1) is likely to fall for each added annotator asymptotically, and even a further 5% (out of 230, resp. 520 errors) would amount to only one or two errors per category – not enough to skew results.

[25] i.e. the percentage of error found only by DanProof.

models, making for a very small environmental footprint.

8. Bibliographical References

Beede, P. and Mulnix, M. W, (2017). Grammar, spelling error rates persist in digital news. *Newspaper Research Journal*, vol. 38, issue 3. https://doi.org/10.1177/0739532917722766

Bick, E. (2006). A Constraint Grammar Based Spellchecker for Danish with a Special Focus on Dyslexics. In: Suominen, Mickael et al. (ed.) *A Man of Measure: Festschrift in Honour of Fred Karlsson on his 60th Birthday.* Special Supplement to SKY Jounal of Linguistics, Vol. 19. pp. 387-396. Turku: The Linguistic Association of Finland

Bick, Eckhard (2011). A FrameNet for Danish. In: *Proceedings of NODALIDA 2011, May 11-13, Riga, Latvia*. NEALT Proceedings Series, Vol 11, pp. 34-41. Tartu: Tartu University Library. ISSN 1736-6305

Bick, E. (2015). DanProof: Pedagogical Spell and Grammar Checking for Danish. In: Galia Angelova, Kalina Bontcheva & Ruslan Mitkov: *Proceedings of RANLP 2015* (Hissar, Bulgaria, 7-9 Sept. 2015). pp. 55-62.

Bick, Eckhard (2023). VISL & CG-3: Constraint Grammar on the Move: An application-driven paradigm. In: Arvi Hurskainen, Kimmo Koskenniemi & Tommi Pirinen (eds.): *Rule-Based Language Technology.* NEALT Monograph Series vol. 2, pp. 112-140. University of Tartu. ISSN 1736-6291

Birn, J. (2000). Detecting grammar errors with Lingsoft's Swedish grammar checker. In Nordgård, Torbjørn (ed.) In : *NODALIDA '99 Proceedings from the 12th Nordiske datalingvistikkdager*, pp. 28-40. Trondheim: Department of Linguistics, University of Trondheim.

Choe, Y. J., Ham, J., Park, K., and Yoon, Y. (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*, pp. 213–227. ACL

De Amorim, R. C. and Zampieri, M. (2013). Effective spell checking methods using clustering algorithms. In *Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP 2013*), pp. 172–178.

Diderichsen, Ph. and Schack, J. (2015). Jagten på den gode og sikre sprogbruger. *Nyt fra Sprognævnet* 2015/3. 1-8.

Downs, B, et al. (2020). KidSpell: A Child-Oriented, Rule-Based, Phonetic Spellchecker. *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020).* pp. 6937–6946. ELRA

Ghosh, S. and Kristensson, P. O. (2017). Neural networks for text correction and completion in keyboard decoding. *CoRR*, abs/1709.06429, http://arxiv.org/abs/1709.06429

Kristensen, L.B., Ibholt, T.B. and Nielsen, A.P. (2007). Avisernes fejl er en gammel nyhed. *Mål & Mæle 30(3)*. pp. 7–11.

Näther, M. (2020). An In-Depth Comparison of 14 Spelling Correction Tools on a Common Benchmark. In *Proceedings of LREC 2020*, pp. 1849–1857. European Language Resource Association (ELRA).

Rathje, M., Schack, J., Blom, J.N., and Bick, E. (2023). Stavefejl i aviserne 1999 og 2019. *Nyt fra Sprognævnet*, 2023/2 (*oktober).* Dansk Sprognævn.

Starlander, M. and Popescu-Belis, A. (2002). Corpus-based Evaluation of a French Spelling and Grammar Checker. In *Proceedings of LREC 2002*, May 29-31, 2002, Las Palmas, Spain.

Sahu, S. et al. (2020). Evaluating performance of different grammar checking tools. *International Journal of Advanced Trends in Computer Science and Engineering.* Volume 9 No.2, March - April 2020. pp. 2227 – 2233

Singh, S. P., Kumar, A., Singh, L., Bhargava, M., Goyal, K., and Sharma, B. (2016). Frequency-based spell checking and rule-based grammar checking. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, pp. 4435–4439. IEEE.

Tetreault, J. R., Sakaguchi, K., and Napoles, C. (2017). JF-LEG: A fluency corpus and benchmark for grammatical error correction. In *Proceedings of EACL 2017*, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers, pp. 229–234.

Wiechetek, L, Moshagen. S.N., Gaup, B. and Omma, Th. (2019). Many shades of grammar checking – launching a Constraint Grammar tool for North Sámi. In: *Proceedings of the NoDaLiDa 2019 Workshop on Constraint Grammar - Methods, Tools and Applications*, NEALT Proceedings Series 33:8, pp. 35–44

Wiechetek, L., Pirinen, F., Hämäläinen, M. and Argese, Ch. (2021). Rules Ruling Neural Networks - Neural vs. Rule-Based Grammar Checking for a Low Resource Language. In: *Proceedings of Recent Advances in Natural Language Processing (RANLP 2021)*. pp.1526–1536