The WikEd Error Corpus: A Corpus of Corrective Wikipedia Edits and its Application to Grammatical Error Correction

Roman Grundkiewicz and Marcin Junczys-Dowmunt

Faculty of Mathematics and Computer Science
Adam Mickiewicz University in Poznań
ul. Umultowska 87, 61-614 Poznań, Poland
{romang,junczys}@amu.edu.pl

Abstract. This paper introduces the freely available WikEd Error Corpus. We describe the data mining process from Wikipedia revision histories, corpus content and format. The corpus consists of more than 12 million sentences with a total of 14 million edits of various types.
As one possible application, we show that WikEd can be successfully adapted to improve a strong baseline in a task of grammatical error correction for English-as-a-Second-Language (ESL) learners' writings by 2.63%. Used together with an ESL error corpus, a composed system gains 1.64% when compared to the ESL-trained system.

Keywords: Error corpus, Wikipedia revision histories, grammatical error correction

1 Introduction

Machine learning approaches in the field of natural language processing are data-hungry. In the ideal case, large and diversified data sets are available that can be used directly or easily adapted to the investigated problem. An example where large amounts of data can be beneficial is automated grammatical error correction for English-as-a-second-language (ESL) learners.

Although some types of errors, for instance subject-verb mistakes can be corrected using heuristic rules, others, like preposition errors, are difficult to correct without substantial amounts of corpus-based information [10]. The above is especially true when Statistical Machine Translation (SMT) toolkits are applied as error correction systems [15]. Compared to multilingual translation corpora which today are plentiful or can be easily collected, genuine error corpora are not easy to come by. If copyright and licensing issues are taken into account as well, the number of resources becomes very scarce.

In this paper, we introduce the — to our knowledge — largest free corpus of corrective edits available for the English language: the WikEd Error Corpus, version 0.9. This corpus consists of edited sentences extracted from Wikipedia revisions, and as such inherits the user-friendly CC BY-SA 3.0 license of the original resource.

In contrast to other works that use Wikipedia to build various NLP resources [13, 21, 3], we processed the entire English Wikipedia revision history[1] and gathered ca. 12 million sentences with annotated edits. Possible application include, but are not limited to, sentence paraphrasing, spelling correction, grammar correction, etc. Both, the WikEd Error Corpus and the tools used to produce it have been made available for unrestricted download[2].

In the next section we describe related work in the domain of error corpora collection. Section 3 presents our language-independent method of edit operation mining from Wikipedia's revision histories and contains descriptions of the collected data, error types, and formats. In Section 4, we demonstrate the usefulness of WikEd to automated ESL grammatical error correction: an SMT-based system is adapted for ESL error correction. Unlike in numerous previous works, we do not restrict ourselves to only a few chosen error types, but attempt a full correction as it has been introduced in this year's CoNLL Shared Task [17]. Finally, we conclude in Section 5 with comments on planned improvements of the WikEd Error Corpus.

2 Related Work

While reviewing related work, we restrict ourselves to approaches to error corpora gathering. For a review of the field of grammatical error correction, we refer the reader to Leacock et al. [10], for the current state-of-the-art, we recommend the proceedings of the 2013 and 2014 CoNLL Shared Tasks [18, 17].

Three main approaches to gathering error corpora are present in literature: manual annotation of students' writings, artificial errors generation within well-formed sentences, and the extraction of errors and their corrections from edit histories. A fourth possibility are social networks for language learners.

2.1 Learner's Corpora

As noted by Leacock et. al [10], even if large quantities of students' writings are produced and corrected every day, only a small number of them is archived in electronic form. Most of the available error-annotated corpora has been created from ESL learners' writings. Examples are the NUS Corpus of Learner English [5] (NUCLE), the dataset of FCE scripts[3] extracted from the Cambridge Learner Corpus, and the International Corpus Network of Asian Learners of English[4].

They are usually small, a few hundreds sentences. NUCLE is a notable exception, but for machine learning approaches even the ca. 50,000 sentences from NUCLE are a rather small resource. It is also worth noting that errors made by learners differ from errors made by native-speakers, therefore, the use of ESL

[1] Wikipedia database dump from January 2nd, 2014: `http://dumps.wikimedia.org/enwiki/20140102/`

[2] `http://romang.home.amu.edu.pl/wiked/wiked.html`

[3] `http://ilexir.co.uk/applications/clc-fce-dataset/`

[4] `http://language.sakura.ne.jp/icnale/`

corpora for the correction of native speaker errors may be limited, and vice versa[5].

2.2 Artificial Errors

One proposed solution to overcome data sparseness is the creation of artificial data. In the case of artificial error corpora, grammatical errors are introduced by random substitutions, insertions, or deletions according to the frequency distribution observed in seed corpora.

Brocket et al. [1] introduce mass/count noun errors with hand-constructed rules. Wagner et al. [19] produce ungrammatical sentences based on an error analysis carried out on a corpus formed by roughly 1,000 error-annotated sentences. Foster and Andersen [6] introduce *GenERRate*, a tool for the production of artificial errors that imitate genuine errors from two data sets: a grammatical corpus and a list of naturally-occuring errors. Yuan and Felice [20] extracted lexical and part-of-speech patterns for five types of errors from NUCLE and applied them to well-formed sentences.

Admittedly, artificial error generation is an efficient and economic way to increase the size of training datasets, but there are drawbacks. The diversification of errors in such corpora can be lower due to small set of real seed data. For specific error types it may be difficult to create descriptive patterns that can be applied to well-formed sentences. Furthermore, it has been reported that artificial data can be less suited for evaluation purposes [21].

2.3 Text Revision Histories

An alternative solution consists in the extraction of errors from text revision histories. The most frequently used are Wikipedia revisions.

Miłkowski [14] proposes the construction of error corpora from text revision histories based on the hypothesis that the majority of frequent minor edits are error corrections. A Polish corpus of errors automatically extracted from Wikipedia revisions has been created by Grundkiewicz [7]. To distinguish error corrections from unwanted edits and to determine error categories the author used hand-written rules.

Wikipedia revisions have been used for the creation of sentence paraphrase corpora by Max and Wisniewski [13], real-word spelling error correction by Zesch [21] and preposition error correction by Cahill et al. [3]. Cahill et al. confirm that data from Wikipedia is useful for both, training a correction system and creating artificial data. This research is the closest to our work, but focuses only on prepositions, whereas we perform experiments on a much larger scale and cover all error types.

The main advantage of Wikipedia-extracted data sets is their size, but there are also disadvantages, for instance Wikipedia's encyclopedic style and an abundance of vandalism.

[5] We show that this is not necessarily true.

2.4 Social Networks for Language Learners

Probably the best resource for language errors has made a very recent appearance in the form of social networks for language learners, an example being Lang-8.com. Learners with different native languages correct each others texts based on their own native-language skills. See Section 4.3 for more information. However, this resources are not free for all purposes, special license agreements are required.

3 The WikEd Error Corpus

In this section we describe our method of edit extraction from Wikipedia revisions which leads to the creation of the WikEd Error Corpus, version 0.9.

3.1 Extracting Edits from Wikipedia

Wikipedia dumps with complete edit histories are provided in XML format[6]. Similarly to Max and Wisniewski [13] and Grundkiewicz [7], we iterate over each two adjacent revisions of every Wikipedia page, including articles, user pages, discussions, and help pages. To minimize the number of unwanted vandalism, we skip revisions and preceding revisions if comments contain suggestions of reversions, e.g. *reverting after (...)*, *remove vandalism*, *undo vandal's edits*, *delete stupid joke*, etc. This is done by a few hand-written rules involving regular expressions.

Next, we remove markup[7] from each article version and split texts into sentences with the NLTK toolkit[8]. Pairs of edited sentences are identified with the Longest Common Subsequence algorithm (LCS) [12]. Edits consisting of additions or deletions of full paragraphs are disregarded.

Two edited sentences[9] s_i and s_j are collected if they meet several surface conditions

- the sentence length is between 2 and 120 tokens,
- the length difference is less than 5 tokens,
- the relative token-based edit distance $\mathrm{ed}(s_i, s_j)$ with respect to the shorter sentence is smaller than 0.3.

The threshold values in the above restrictions were chosen experientially. The relative token-based edit distance is defined as:

$$\mathrm{ed}(s_i, s_j) = \frac{\mathrm{dist}(s_i, s_j)\min(|s_i|, |s_j|)}{\log_b \min(|s_i|, |s_j|)},$$

[6] `http://dumps.wikimedia.org/`
[7] `http://medialab.di.unipi.it/wiki/Wikipedia_Extractor`
[8] `http://nltk.org/`
[9] In the remainder of this paper we will refer to two corresponding edited fragments as sentences, even if they are not well-formed.

where $\mathrm{dist}(s_i, s_j)$ is the token-based Levenshtein edit distance [11], $|s|$ is the length of the sentence s in tokens, and the logarithm base b is empirically set to 20. This formula implies that the longer the sentence is, the more edits are allowed, but it prevents the acceptance of too many edits for long sentences.

3.2 Collected Corrective Edits

At this stage, 12,130,508 pairs of edited sentences from the English version of Wikipedia have been collected. The most useful edits include:

- spelling error corrections:
  `You can use rsync to [-donload-] {+download+} the database .`,
- grammatical error corrections:
  `There [-is-] {+are+} also [-a-] two computer games based on the movie .`,
- stylistic changes:
  `[-Predictably , the-] {+The+} game ended [-predictably-] when she crashed her Escalade...`,
- sentence rewordings and paraphrases:
  `These anarchists [-argue against-] {+oppose the+} regulation of corporations .`,
- encyclopaedic style adjustments:
  `A [-local education authority-] {+Local Education Authority+} ( LEA ) is the part of a council in England or Wales.`

The WikEd corpus contains also less useful edits for grammatical error correction task, e.g.:

- time reference changes:
  `The Kiwi Party [-is-] {+was+} a New Zealand political party formed in 2007 .`,
- information supplements:
  `Aphrodite is the Greek goddess of love {+, sex+} and beauty .`,
- numeric information updates:
  `In [-May 2003-] {+August 2004+} this percentage increased to [-62-] {+67+}% .`,
- item additions/deletions to/from bulleted lists:
  `Famous Bronxites include {+Regis Philbin ,+} Carl Reiner , Danny Aiello...`,
- amendments of broken MediaWiki's markups:
  `The bipyramids are the [-[ [ dual polyhedron |-] dual polyhedra [-[ [-] of the prisms.`,
- changes made by vandals:
  `David Zuckerman is a writer and [-producer-] {+poopface+} for television shows.`

The total number of edits is 16,013,830 among which 3,273,862 (20,44%) are deletions and 4,829,019 (30.16%) insertions. The most frequently occurring edits are presented in Table 1.

Table 1: 30 most frequent edits in the WikEd 0.9 corpus.

Edits	Freq.	Edits	Freq.	Edits	Freq.
`ins(")`	667,098	`ins(a)`	45,870	`ins(and)`	28,518
`ins(,)`	348,341	`ins(')`	41,473	`del(of)`	26,257
`del(")`	226,854	`del(.)`	41,161	`sub(a,an)`	24,626
`del(,)`	158,324	`sub(is,was)`	40,062	`ins(was)`	23,670
`ins(.)`	138,322	`sub(',")`	37,236	`del(])`	22,443
`del('s)`	80,669	`del(')`	36,051	`sub(was,is)`	21,372
`ins(the)`	79,708	`del())`	34,401	`ins(()`	20,079
`del(the)`	61,999	`del(persons)`	33,773	`del(a)`	19,615
`ins())`	60,852	`ins(The)`	32,819	`ins(in)`	18,651
`ins(< br >)`	51,802	`sub(it,its)`	31,171	`ins(is)`	18,647

3.3 Filtering

As shown by Grundkiewicz [7], sentences with potentially unwanted edits, e.g. updates of bulleted list, amendments of MediaWiki markup, and vandalism can be effectively filtered out using heuristic rules. For example, all pairs of sentences s_i and s_j that satisfy the following conditions can be disregarded:

- Either the sentence s_i or s_j consists of a vulgar word (determined by the list of vulgarisms) or a very long sequence of character with no spaces (e.g. produced by random keystrokes).
- Any of the sentences s_i or s_j contains fragments of markup, e.g. `<ref>`, `<br>` or `[http:`.
- All edits concern only changes in dates or numerical values.
- The only edit made consists of removing a full stop or semicolon at the end of the sentence s_i.
- The ratio of non-words tokens in s_j to word tokens is higher than a given threshold (we used 0.5).

In the end, 1,775,880 (14.63%) pairs of sentences are marked as potentially harmful, but not removed. For instance, vandalized entries may be useful for various tasks by themselves.

3.4 Corpus Format

It is our intention to release the WikEd Error Corpus in a machine-friendly format. We chose a representation based on GNU `wdiff` output[10] extended by comments including meta-data. For example, for a sentence *This page lists links about ancient philosophy.* with the following two edits: insertion of *some* at third position and substitution of *about* with *to*, the WikEd entry corresponds to:

[10] https://www.gnu.org/software/wdiff/manual/wdiff.html#wdiff

`This page lists {+some+} links [-about-] {+to+} ancient philosophy.`

Meta-data consists of:

- the revision id, accompanying comment, and revision timestamp,
- the title and id of the edited Wikipedia page,
- the name of the contributor or IP address if it is an anonymous edition.

All sentences preserve the chronological order of the original revisions.

4 Application to ESL Error Correction

In the second part of the paper we examine the usefulness of the WikEd Error Corpus in an automated ESL error correction scenario. Despite the fact, that WikEd is not an English-as-a-Second-Language (ESL) learners' error corpus — although it may contain a substantial number of errors contributed by non-native English users — we demonstrate that it is possible to select mistakes in such a way that an ESL error correction system can benefit from WikEd.

4.1 Task Description

We take advantage of the training data published during the CoNLL-2014 Shared Task on Grammatical Error Correction [17]. The aim of the shared task was to automatically correct essays written by Singaporean ESL learners. Training data has been made available in form of the previously mentioned NUCLE corpus [5]. NUCLE consists of 1,414 essays (57,151 sentences) which cover a wide range of topics, such as environmental pollution and health care. The sentences have been corrected by professional English instructors and annotated with 44,385 corrections in 28 error categories, such as article or determiner errors, wrong collocation or idiom, noun number errors, etc.

System performance is measured by the MaxMatch (M^2) metric [4] which computes the F-score for the proposed corrections against a gold standard that has been similarly annotated as NUCLE. It is not necessary to correctly classify error types, only the text of the correction is compared. In this paper, we use the test set (ST-2013) from the previous edition of the CoNLL shared task for evaluation. It has been made available as training data for the current shared task and contains annotation for all 28 error types. Apart from testing on ST-2013, we also report results for 4-fold cross validation on NUCLE.

4.2 System Description

Due to our background in statistical machine translation and the rising popularity of grammatical error correction by SMT, we decided to use the Moses [9] toolkit to built our demonstration system. Our baseline is a re-implementation of an intermediate system from Junczys-Dowmunt and Grundkiewicz [8] which

is labeled by the authors as NUCLE+ CCLM. This system uses NUCLE as the sole parallel training data. It also adds a web-scale language model estimated from English CommonCrawl data made available by Buck et al. [2]. For the 28 error categories, the baseline achieves $F_{0.5}$=27.43%.

We can assume that this is a strong baseline. For the previous 5 error-type task from the CoNLL-2013 Shared Task the same system achieves F_1=29.84% (CoNLL-2013-ST used F_1, CoNLL-2014-ST changed to $F_{0.5}$). Had it taken part in the task, it would have ranked on second place among 17 teams, only 1.36% below the winning system and over 4% higher than the next best submission.

During training, 4-fold cross validation has to be adjusted to accommodate parameter tuning as is common practice in SMT. This results in a testing/tuning scheme labeled 4×2-fold cross validation (4×2-CV). The original test sets from 4-fold cross validation are divided into two halves and both are used for cross tuning and testing. This results in four training steps with two testing/tuning steps each. The eight tuned parameter weight vectors are averaged and the centroid vector is used to translate ST-2013 with a translation model estimated from the complete NUCLE data.

In the grammatical error correction scenario where source and target phrases are often identical or similar, it might be useful to inform the decoder about the differences in a phrase pair. Similarly to Junczys-Dowmunt and Grundkiewicz [8] we extend translation models with a word-based Levenshtein distance feature [11] that captures the number of edit operations required to turn the source phrase into the target phrase. Each phrase pair in the phrase table is scored with $e^{d(s,t)}$ where d is the word-based distance function, s is the source phrase, t is the target phrase. The exponential function e^x is used because Moses scores translations e of string f by a log-linear model

$$\log p(e|f) = \sum_i \lambda_i \log(h_i(e, f)),$$

where h_i are feature functions and λ_i are feature weights. That way the model score include the total numer of edits in a sentence counted by the Levenshtein distance feature for individual phrases pairs. This feature should also be helpful for reducing noise in the translation output. During evaluation, we refer to this component as "LD".

4.3 True ESL Error Data

We also compare our data to a true ESL corpus. Mizumoto et al. [16] published[11] a list of learners' corpora that were scraped from the social language learning site Lang-8 (`http://lang-8.com`). Version 1.0 is available for academic purposes, commercial applications require special licenses from the copyright owner. Newer versions (2.0) require special license agreements for any usage.

We collect all entries from "Lang-8 Learner Corpora v1.0" with English as the learned language, we do not care about the native language of the user.

[11] `http://cl.naist.jp/nldata/lang-8`

Table 2: The comparison of the WikEd 0.9 and Lang-8 NAIST corpora.

Statistics	WikEd 0.9	+Select	L8-NAIST	+Select
sentences	12,130,508	—	2,567,964	—
tokens (source side)	292,570,716	294,965,241	28,506,516	34,351,819
edits	16,013,830	5,327,293	3,408,834	1,066,690
sentences with ≥1 edits	91.79%	32.62%	53.86%	28.15%
edits per sentence	1.32	0.44	1.33	0.42

Only entries for which at least one sentence has been corrected are taken into account. Sentences without corrections from such entries are treated as error-free and mirrored on the target side of the corpus. Eventually, we obtain a corpus of 2,567,969 sentence pairs with 28,506,540 tokens on the uncorrected source side. We call this resource "L8-NAIST". The comparison with the WikEd 0.9 corpus is presented in Table 2.

4.4 Error Selection

As mentioned before, the WikEd Error Corpus is not an ESL error corpus and may contain a very different type of errors from those made by language learners. We try to mitigate this by selecting errors that resemble mistakes from NUCLE, other errors are replaced by their corrections.

For each pair of uncorrected and corrected sentences from NUCLE, we compute a sequence of deletions and insertions with the LCS algorithm that transform the source sentence into the target sentence. Adjacent deleted words are concatenated to form a phrase deletion, adjacent inserted words result in a phrase insertion. A deleted phrase followed directly by a phrase insertion is interpreted as a phrase substitution. Substitutions are generalized if they consist of common substrings, again determined by the LCS algorithm, that are equal to or longer than three characters. We encode generalizations by the regular expression `(\w{3,})` and a back-reference, e.g. `\1`.

Patterns can contain multi-word strings, e.g. `sub((\w{3,}) is,\1s are)` models a case of subject-verb agreement. Sometimes, more than one generalization is possible, e.g. `sub((\w{3,})-(\w{3,}),\1\2)`. Table 3 contains the some of the most frequent patterns extracted from NUCLE for all 28 error types. The table includes also the most frequent error categories matching the pattern. A frequency threshold is defined at 5, patterns that occur less often are discarded, in the end 666 patterns remain.

Next, we perform the same computation for sentence pairs from WikEd. Edits that result in patterns from our list are not modified and remain in the data, for all other edits, the selected correction is applied to the source sentence. Error types not covered by the patterns thus disappear. Noise like vandalism is either removed or reduced to identical sentences on both sides for the training corpus. In both cases this cannot harm our systems. Eventually, 3,957,547 (32,62%) sentences remain that still contain edit pattern. We keep all sentences with surviving

Table 3: 14 most frequent patterns extracted from NUCLE 3.0

Pattern	Freq.	Categories with Freq.
sub((\w{3,}),\1s)	2864	Nn(2188) SVA(395) Wform(146)
ins(the)	2494	ArtOrDet(2424)
del(the)	1772	ArtOrDet(1696)
sub((\w{3,})s,\1)	1317	Nn(651) SVA(263) Wform(141) Rloc-(92)
ins(,)	971	Mec(733) Srun(196)
ins(a)	679	ArtOrDet(646)
sub((\w{3,}),\1d)	300	Vt(112) Vform(105) Wform(62)
del(,)	266	Mec(175) Rloc-(83)
sub((\w{3,}),\1ed)	252	Vt(138) Vform(75) Wform(29)
ins(an)	246	ArtOrDet(234)
del(of	222	Prep(202)
sub(is,are)	219	SVA(198)
del(.)	205	Rloc-(135) Mec(60)
sub((\w{3,})d,\1)	202	Vt(109) Wform(46) Vform(28) Rloc-(11)

errors and randomly select sentences without edits to be kept as well. The final parallel corpus consists of 4,703,353 sentence pairs. Two versions are used in our experiments: the error selected corpus which is labeled "WikEd+Select" and a second version consisting of the same sentences but with all errors present (a proper subset of the unprocessed WikEd), this version is denoted as "WikEd".

Error selection is also applied to L8-NAIST, resulting in L8-NAIST+Select, all sentences remain in this resource.

4.5 Results

Table 4 contains results for our experiments with WikEd and L8-NAIST. Unadapted WikEd used as parallel training data lowers the results drastically, which is not surprising, many edits may be different in style and scope and considered harmful for the ESL-based NUCLE. Adding the LD feature makes it even worse.

Error selection changes the picture. Only errors similar to NUCLE data remain in the training corpus. Results improve for both, NUCLE cross-validation (4×2-CV) and the unseen test set ST-2013. The latter excludes the possibility of overfitting to NUCLE due to error selection as might be postulated based on cross-validation results alone. Adding LD to the error-selected version of WikEd leads to further gains in both cases. Eventually, improvements of 2.24% and 2.63% $F_{0.5}$-score over the baseline can be observed.

We perform the same experiments with the ESL corpus L8-NAIST. It is, of course, not surprising that the in-domain L8-NAIST performs much better than WikEd. It should however be noted that the significant performance improvements stem from our error selection procedure and the Levenshtein distance feature. Final results achieve 4.21% and 6.72% over the baseline.

Table 4: Evaluation for grammatical error correction task

System	4×2-CV	ST-2013
NUCLE+CCLM	22.19	27.43
+WikEd	18.96	26.12
+LD	18.21	23.63
+Select	23.80	29.49
+LD	**24.33**	**30.06**

(a) WikEd Error Corpus 0.9

System	4×2-CV	ST-2013
NUCLE+CCLM	22.19	27.43
+L8-NAIST	23.34	31.20
+LD	24.44	34.06
+Select	25.43	33.89
+LD	**26.40**	**34.15**

(b) Lang-8 Error Corpora 1.0

System	4×2-CV	ST-2013
Joint-Translation	26.01	32.33
Composition	**26.63**	**35.79**

(c) System combination results

For both error corpora, the improvements are due the combined effects of additional parallel data, error selection, and the task-specific LD feature. In order to verify that this is not alone the beneficial effect of the LD feature on data present in NUCLE, we also test NUCLE+CCLM+LD with does not improve the baseline (22.10% and 27.62%).

We also evaluated two simple system combinations for which results are presented in Table 4c:

- "Joint-Translation", which is a single Moses system configured to use both separately tuned phrase tables from the best two systems — NUCLE+CCLM +WikEd+Select+LD and NUCLE+CCLM+L8-NAIST+Select+LD;
- "Composition", which is a chain of NUCLE+CCLM+WikEd+Select+LD and NUCLE+CCLM+L8-NAIST+Select+LD. The output of the first system is corrected a second time by the latter.

In the case of Joint-Translation we see worse results than for the best L8-NAIST-trained system. It seems that the parameter tuning process could not take advantage of the two phrase tables. However, results for "Composition" are significantly better (+1.64%) than for the single systems, both systems tend to correct slightly different errors which results in a more correct composed output. Experiments with other system combination techniques from SMT should be performed in the future.

5 Conclusions and Future Work

With this paper, we introduced the WikEd Error Corpus — a publicly available large corpus of corrective Wikipedia edits. It consists of more than 12 million sentences with a total of 14 million edits of various types.

A certain portion of noisy edits is included, but as was shown in this work it can be adapted to specific tasks when seed data is available. There is nothing that prevents other researchers from tailoring the corpus to their own purposes. Advantages of WikEd are its size and friendly license, and we believe the collected data to be more reliable than artificially created errors.

As demonstrated, despite not being an ESL corpus, WikEd can be successfully adapted to improve a strong baseline in an ESL grammatical error correction task by 2.63%. When used together with an ESL error corpus, a composed system gains 1.64% when compared to the ESL-trained system alone.

Future work should concentrate on better cleaning methods and additional meta-data. Version 1.0 should also see automatically added linguistic knowledge, for instance part-of-speech tagging. An obvious and planned extension is the expansion to other languages and the addition of new sources in the form of other wiki-sites with publicly available revision histories.

References

1. Brockett, C., Dolan, W.B., Gamon, M.: Correcting ESL Errors Using Phrasal SMT Techniques. In: Proceedings of ACL. pp. 249–256. ACL (2006)
2. Buck, C., Heafield, K., van Ooyen, B.: N-gram Counts and Language Models from the Common Crawl. In: Proceedings of LREC (2014)
3. Cahill, A., Madnani, N., Tetreault, J.R., Napolitano, D.: Robust Systems for Preposition Error Correction Using Wikipedia Revisions. In: Proceedings of NAACL: HLT. pp. 507–517. ACL (2013)
4. Dahlmeier, D., Ng, H.T.: Better Evaluation for Grammatical Error Correction. In: Proceedings of NAACL: HLT. pp. 568–572. ACL (2012)
5. Dahlmeier, D., Ng, H.T., Wu, S.M.: 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. pp. 22–31. ACL (2013)
6. Foster, J., Andersen, O.E.: GenERRate: Generating Errors for Use in Grammatical Error Detection. In: Proceedings of the Fourth Workshop on Innovative Use of NLP for Building Educational Applications. pp. 82–90. ACL (2009)
7. Grundkiewicz, R.: Automatic Extraction of Polish Language Errors from Text Edition History. In: Proceedings of TSD. pp. 129–136. Springer (2013)
8. Junczys-Dowmunt, M., Grundkiewicz, R.: The AMU System in the CoNLL-2014 Shared Task: Grammatical Error Correction by Data-Intensive and Feature-Rich Statistical Machine Translation. In: Proceedings of CoNLL: Shared Task. ACL (2014)
9. 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. In: Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions. pp. 177–180. ACL '07, ACL (2007)
10. Leacock, C., Chodorow, M., Gamon, M., Tetreault, J.: Automated Grammatical Error Detection for Language Learners. Morgan and Claypool Publishers (2010)
11. Levenshtein, V.I.: Binary Codes Capable of Correcting Deletions, Insertions and Reversals. Soviet Physics Doklady 10 (1966)

12. Maier, D.: The Complexity of Some Problems on Subsequences and Supersequences. J. ACM 25(2), 322–336 (1978)
13. Max, A., Wisniewski, G.: Mining Naturally-occurring Corrections and Paraphrases from Wikipedia's Revision History. In: Proceedings of LREC (2010)
14. Miłkowski, M.: Automated Building of Error Corpora of Polish. In: Corpus Linguistics, Computer Tools, and Applications — State of the Art, pp. 631–639. Peter Lang (2008)
15. Mizumoto, T., Hayashibe, Y., Komachi, M., Nagata, M., Matsumoto, Y.: The Effect of Learner Corpus Size in Grammatical Error Correction of ESL Writings. In: Proceedings of COLING 2012: Posters. pp. 863–872 (2012)
16. Mizumoto, T., Komachi, M., Nagata, M., Matsumoto, Y.: Mining Revision Log of Language Learning SNS for Automated Japanese Error Correction of Second Language Learners. In: IJCNLP. pp. 147–155 (2011)
17. Ng, H.T., Wu, S.M., Briscoe, T., Hadiwinoto, C., Susanto, R.H., , Bryant, C.: The CoNLL-2014 Shared Task on Grammatical Error Correction. In: Proceedings of CoNLL: Shared Task. ACL (2014)
18. Ng, H.T., Wu, S.M., Wu, Y., Hadiwinoto, C., Tetreault, J.: The CoNLL-2013 Shared Task on Grammatical Error Correction. In: Proceedings of CoNLL: Shared Task. pp. 1–12. ACL (2013)
19. Wagner, J., Foster, J., van Genabith, J.: A Comparative Evaluation of Deep and Shallow Approaches to the Automatic Detection of Common Grammatical Errors. In: Proceedings of EMNLP-CoNLL. pp. 112–121. ACL (2007)
20. Yuan, Z., Felice, M.: Constrained Grammatical Error Correction using Statistical Machine Translation. In: Proceedings of CoNLL: Shared Task. pp. 52–61. ACL (2013)
21. Zesch, T.: Measuring Contextual Fitness Using Error Contexts Extracted from the Wikipedia Revision History. In: Proceedings of EACL. pp. 529–538 (2012)