A Crash Course in Automatic Grammatical Error Correction

COLING'2020 Tutorial

Roman Grundkiewicz [†], Christopher Bryant [‡], Mariano Felice [‡]
rgrundki@inf.ed.ac.uk, {cjb255,mf501}@cl.cam.ac.uk

December 12, 2020

[†] Microsoft / University of Edinburgh [‡] University of Cambridge

Part I. Introduction

About us

Roman

Chris

Mariano

- Working on automatic grammatical error correction since 2013–2014.
- Creating W&I+LOCNESS (Chris, Mariano) and WikEd (Roman) error corpora.
- Organizing CoNLL 2014 (Chris) and BEA 2019 (Chris, Mariano) shared tasks on grammatical error correction.
- Building best systems at CoNLL 2014 (Mariano) and BEA 2019 (Roman) shared tasks.

About the tutorial

Goal: Introduce attendees to recent progress in automatic Grammatical Error Correction (GEC).

- Focus on GEC for English as a Second Language (ESL) learners.
- Low-resource GEC for other languages.

Target audience: Newcomers to the field with machine learning or computational linguistics backgrounds.

Most recent version of slides and list of resources:
`https://github.com/grammatical/coling2020-tutorial`

About automatic grammatical error correction

Growing academic and commercial interest.

$\rightarrow$ 24 participating teams in the BEA 2019 Shared Task

Practical applications as a tool for language learners and native speakers, or possibly a post/preprocessing step for other natural language processing tasks.

Tutorial outline

Part I. Introduction

Part II. Historical and recent approaches

Part III. Data and evaluation

Part IV. Neural grammatical error correction

Part V. Recent and future work

Grammatical error correction (GEC)

*I think, that everybody **deserve** privacy, including famous people.*
*They can **barelly breathing** with all those photographers around them.*
*I don't know why people love **spying** famous people.*
And magazines are full of those things.

$\downarrow$

*I think that everybody **deserves** privacy, including famous people.*
*They can **barely breath** with all those photographers around them.*
*I don't know why people love **spying on** famous people.*
And magazines are full of those things.

Example from the W&I+LOCNESS corpus

Grammatical error correction (GEC)

Task formulation:

- Automatic sequence-to-sequence task.
- Error detection and correction.
- All types or errors, including grammatical, lexical and orthographical errors.
    - In practice, the set of errors is defined by datasets.
    - English as a Second Language (ESL) corpora.

Related tasks: grammatical error detection, spelling correction, essay scoring, style transfer, automatic post-editing, and others.

Challenges

1. Multiple corrections are acceptable.

   Above all, life is more important than {secret→secrets|secrecy|a secret}.
   {In conclude→In conclusion|To conclude}, social media benefit people.

2. Multiple errors may occur in a single sentence.

   19-58% of sentences in ESL corpora contain more than one annotation.

3. Long-distance dependencies, including cross-sentence dependencies.

   A subtle scent of red sweet apples and cinnamon sticks {are→is} present in the wine .

Challenges

4. Some error types are more difficult to correct than others.

   Closed-class error types (e.g. articles) vs. open-class errors.

5. Low frequency of errors.

   Depending on the ESL error corpus, 35-85% sentences contain one or more errors and only 6-15% erroneous words. These numbers are lower for texts written by native speakers.

6. Error types and error distributions vary significantly among writers and datasets.

... and more. Technical challenges include lower performance of NLP tools on non error-free texts, scarcity of annotated data, etc.

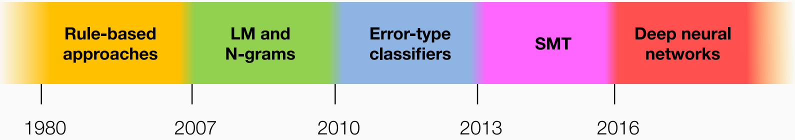
Part II. Historical and recent approaches

Historical and recent approaches

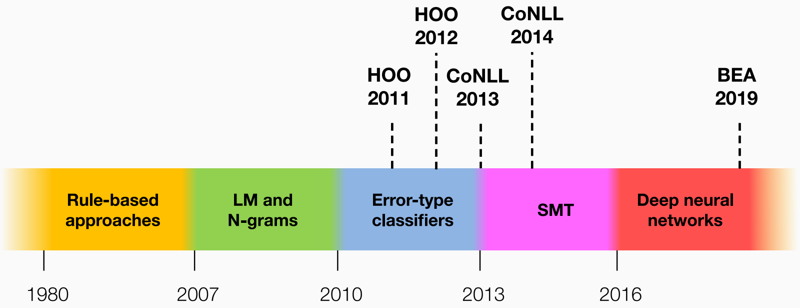
Section overview

1. Rule-based methods
2. Language models and n-gram counts
3. Error-type classifiers
4. Statistical machine translation
5. Deep neural networks
6. Shared tasks

Timeline


Rule-based approaches
LM and N-grams
Error-type classifiers
SMT
Deep neural networks
1980
2007
2010
2013
2016

Timeline


HOO 2012
CoNLL 2014
HOO 2011
CoNLL 2013
BEA 2019
Rule-based approaches
LM and N-grams
Error-type classifiers
SMT
Deep neural networks
1980
2007
2010
2013
2016

Rule-based methods

- String-matching rules, e.g. The Writer's Workbench (Macdonald et al., 1982).
- No context.

  people is → people are

  Recruiting the right people is essential for success.
- Regular expressions:

  /DT_a NNS/

  She bought a cars.
- Wordlists:

  eated → ate

  acomodation → accommodation

Rule-based methods

- Early '90s: basic linguistic analysis and hand-crafted rules.
- ALEK (Chodorow and Leacock, 2000; Leacock and Chodorow, 2003), GRANSKA (Domeij et al., 2000) and ESL Assistant (Gamon et al., 2009).
- ALEK example:

  Noun number: `/DT_a NNS/`
  if not `/DT_a NNS NN/` (e.g. *a systems analyst*)
  or if original frequency $<$ correction frequency

Rule-based methods

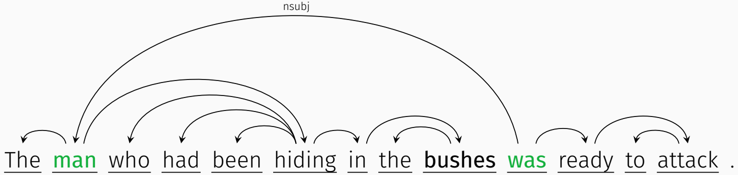
- Microsoft Word: parsing and phrase structure rules.

  I don't have nothing.

```
FORMULA1 (+Pres +Proposition)
└ OpDomain--FORMULA2
    └ L_Sub --- NOMINAL1
        └ SemHeads -- I1
    └ L_Obj --- NOMINAL2 (+ExstQuant)
        └ SemHeads--nothing1
    └ SemHeads--have1
└ SemHeads--not1
```

Rule-based methods

- Full syntactic analysis with parsers and sophisticated grammars (Heidorn et al., 1982; Richardson and Braden-Harder, 1988; Arppe, 2000).
- Typical use cases: agreement errors and sentence fragments.



Rule-based methods

- Many grammar checking products use handcrafted rules (AbiWord, After the Deadline, LanguageTool).
- High precision but depend on the accuracy of tools (e.g. parsers).
- E.g. LanguageTool:

  Wrong usage of modal verbs in questions.

  Does someone can reproduce what I described before?

```
<rule id="DOES_XX_CAN" >
  <pattern>
    <token postag="SENT_START"/>
    <marker>
      <token regexp="yes">Do(es)?</token>
      <token postag="NN:U|PRP\$|PRP" postag_regexp="yes"/>
      <token postag="MD"><exception>need</exception></token>
    </marker>
  </pattern>
</rule>
```

Rule-based methods

- Late '90s-2000s: rules automatically extracted from corpora.
- Spell-checking (Mangu and Brill, 1997), hyphen usage (Rozovskaya et al., 2011; Cahill et al., 2013a).
- General learner English: Write&Improve (Andersen et al., 2013; Yannakoudakis et al., 2018):
  - Patterns of unigrams, bigrams and trigrams extracted from the Cambridge Learner Corpus (CLC).
  - N-grams which have been annotated as incorrect at least five times and ninety per cent of the times they occur.
  - E.g. informations → information
    in the other hand → on the other hand
    busstop → bus stop
    only could → could only

Rule-based methods

Strengths and weaknesses

- 👍 Can be as simple or complex as required
- 👍 Usually very precise
- 👍 Easy to interpret
- 👎 Unsuitable for complex error types (e.g. semantic errors)
- 👎 Require language-specific knowledge
- 👎 Hard to scale and maintain

Language models and n-gram counts

- Frequency as a proxy for grammaticality.
- For **detection**: judge grammaticality directly from LM probabilities (Okanohara and Tsujii, 2007; Wagner et al., 2007; Heilman et al., 2014; Lin and Chen, 2015).

We work from home know.	-42.379
We work from home now.	-30.573

Language models and n-gram counts

- For **correction**: predict words or validate corrections. Correction candidates are taken from predefined sets or generated 'on the fly'. The corrected version must score higher than the original, often based on a threshold (Bergsma et al., 2009; Islam and Inkpen, 2011; Xie et al., 2015; Bryant, 2018).

I am looking forway to see you soon. -2.71

{**forward:** −1.80, Norway: −2.36, foray: −2.70}

↓

I am looking forward to see you soon. -1.80

{**seeing:** −1.65, saw: −2.85, sees: −2.09}

↓

I am looking forward to seeing you soon. -1.65

Language models and n-gram counts

- LMs often used for ranking correction hypotheses, e.g. for SMT (Boroş et al., 2014; Felice et al., 2014; Yuan et al., 2016).

Src	*There are some informations you have asked me about .*	-53.581
Ref	*There is some information you have asked me about .*	-46.672
1	There is some information you have asked me about .	-46.672
2	There is some information you asked me about .	-46.730
3	There is some information you have asked me .	-48.843
4	There are some information you have asked me about .	-49.011
5	There are some information you asked me about .	-49.114
6	There are some information you have asked me .	-51.203
7	There are some information you asked me for .	-51.484
8	There are some information you have asked me for .	-51.723
9	There are some information you have asked me about .	-54.076
10	There are some information you have asked me about it .	-54.655

Language models and n-gram counts

- LMs rarely used on their own but as part of bigger systems.
- Initially trained on the target side of an error-corrected corpus but then moved to bigger general-purpose LMs.
- Some attempts at using the web as a corpus (Fallman, 2002; Hermet et al., 2008; Tetreault and Chodorow, 2009; Gamon and Leacock, 2010).

$$\begin{aligned} &\text{arrived at} &\approx 146,000,000 \text{ hits}\\ &\text{arrived in} &\approx 90,800,000 \text{ hits}\\ &\text{arrive to} &\approx 22,900,000 \text{ hits} \end{aligned}$$

Language models and n-gram counts

Strengths and weaknesses

- 👍 Only require (lots of) native/unannotated text
- 👍 Can target all error types
- 👍 Easy to implement
- 👍 Versatile
- 👎 Probability is not grammaticality
  I am at home vs *I was at home*
- 👎 Rare/unseen words: paraklausithyron, covfefe
- 👎 Cannot handle long-range dependencies well
- 👎 Require confusion sets for correction that can be hard to generate.

Error-type classifiers

- Classifiers were the earliest machine learning approaches.
- Receive a number of features representing the context of a word or phrase and output a predicted class (correction).

  If original = prediction $\rightarrow$ leave unchanged.

  If original $\neq$ prediction $\rightarrow$ correct.
- Most common error types have limited confusion sets so can be turned into a classification task, e.g. articles and prepositions.
- E.g. article classifier:
  - For each noun phrase:

Features	*Classes*
previous 3-gram	no article
next 3-gram	definite article
head noun	indefinite article
head noun word embedding	
...	

Error-type classifiers

Selection vs. *correction* (Rozovskaya and Roth, 2011).

- Selection: predict the class ***without*** the source word as a feature.
- Correction: predict the class ***with*** the source word as a feature.
- *Correction* can model typical confusions and lead to better performance.

Error-type classifiers

Particularly good for:

- **articles** (Lee, 2004; De Felice and Pulman, 2008; Gamon, 2010; Sakaguchi et al., 2012; Rozovskaya et al., 2013),
- **prepositions** (De Felice and Pulman, 2008; Dahlmeier and Ng, 2011; Quan et al., 2012; Cahill et al., 2013b; Jia et al., 2013; Zhang and Wang, 2014),
- **noun number** (Berend et al., 2013; Jia et al., 2013; Rozovskaya et al., 2013; van den Bosch and Berck, 2013; Rozovskaya et al., 2013; Yoshimoto et al., 2013),
- **subject-verb agreement and verb forms** (Jia et al., 2013; van den Bosch and Berck, 2013; Rozovskaya et al., 2013),
- a few others (Rozovskaya et al., 2011, 2014; Wang et al., 2014).

Error-type classifiers

- Correction of open-class words is trickier.
- Need to restrict the output to a finite set of alternatives.
- Lists of candidates can be compiled automatically.
- E.g. Wu et al. (2010) suggest the most appropriate verb in verb-object combinations from 790 verbs.
- Not very efficient so rarely used.

Error-type classifiers

Common classification techniques:

- Naive Bayes
- Logistic regression
- Maximum entropy models
- Support Vector Machines
- ...

Training data:

- Native text (correct)
- Non-native error-annotated data
- Artificial data
- Hybrid datasets

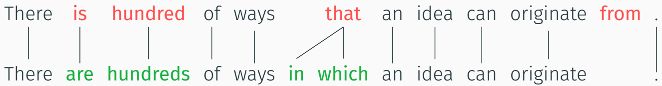
Error-type classifiers

Strengths and weaknesses

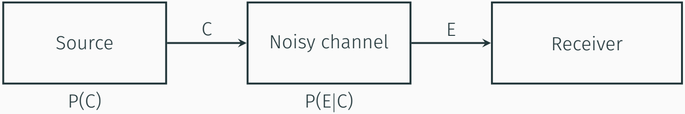
- 👍 More flexible than rules
- 👍 Can be trained on native, non-native or hybrid data
- 👎 Feature engineering can be complicated
- 👎 Better for closed-class error types
- 👎 Typically target single error types
- 👎 Word insertions can be tricky
- 👎 Classifier order matters

Statistical Machine Translation

- GEC can be viewed as a translation from "incorrect" into "correct" English.



- SMT is inspired by the noisy channel model (Shannon, 1948):



- Requires a parallel corpus of original $\rightarrow$ corrected sentences (error annotation is not required).
- Artificial data used if real data is insufficient.

Statistical Machine Translation

1. Align sentences at the word level.
2. Extract phrase mappings into a phrase table.
3. Generate translations using the phrase table and a language model (i.e. decoding).

$$\hat{C} = \arg\max_C P(C|E) = \arg\max_C \frac{P(E|C)P(C)}{P(E)} = \arg\max_C P(E|C)P(C)$$

Statistical Machine Translation

Src Let 's discuss about this informations .

<table>
<tr><td>Let</td><td>'s</td><td>discuss</td><td>about</td><td>this</td><td>informations</td><td>.</td></tr>
<tr><td colspan="2">Lets</td><td>talk</td><td>over</td><td>the</td><td>information</td><td>?</td></tr>
<tr><td colspan="2">Let 's</td><td colspan="2">discuss</td><td>the</td><td>information</td><td>!</td></tr>
<tr><td colspan="2"></td><td colspan="2">talk about</td><td colspan="2">this information</td><td></td></tr>
<tr><td colspan="2"></td><td colspan="2"></td><td colspan="2">these informations</td><td></td></tr>
</table>

Hyp Let 's discuss this information .

Statistical Machine Translation

- Brockett et al. (2006) trained an SMT system to correct noun countability errors using artificial data.
- SMT was a popular approach among participants in the CoNLL-2014 GEC shared task and an integral part of two of the top systems (Felice et al., 2014; Junczys-Dowmunt and Grundkiewicz, 2014).

Statistical Machine Translation

Related approaches

- Correction using round-trip translations (Hermet and Désilets, 2009; Madnani et al., 2012).

Src	*I used to **going** to **camp wich** is **situeded on a** seaside.*	(English)
	↓	
Trans	*Andavo al campo che si trova in riva al mare.*	(Italian)
	↓	
Hyp	*I used to go to the campsite which is located by the sea.*	(English)

- Beam-search decoding using output from other components (Park and Levy, 2011; West et al., 2011; Dahlmeier and Ng, 2012a; Buys and van der Merwe, 2013; Wilcox-O'Hearn, 2013).

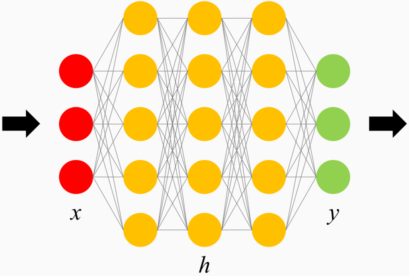
Statistical Machine Translation

Strengths and weaknesses

- 👍 Corrects all error types simultaneously
- 👍 Handles interacting errors
- 👍 Works at the phrase level, not individual words
- 👍 Requires no feature engineering or linguistic knowledge
- 👍 Easy to train for other languages provided data is available
- 👎 Requires lots of parallel training data
- 👎 Out-Of-Vocabulary words (OOV)
- 👎 Hard to customise

Deep neural networks

- General models that map an input x to an output y via a number of hidden states h.

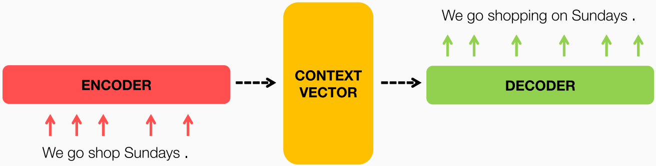


- Their success in other tasks inspired its use in GEC.
- Different architectures and applications under the *deep learning* umbrella.

Deep neural networks

Neural Machine Translation

- Same concept as SMT but with neural networks.
- Sequence-to-sequence model based on the *encoder-decoder* framework.



- Increasingly popular given its effectiveness (Xie et al., 2016; Yuan and Briscoe, 2016; Ji et al., 2017; Chollampatt and Ng, 2018a; Grundkiewicz and Junczys-Dowmunt, 2018a; Grundkiewicz et al., 2019; Chen et al., 2020)

Deep neural networks

Other neural approaches

- Correcting article errors using Convolutional Neural Networks (CNNs) (Sun et al., 2015)
- Sequence labelling for error *detection* using Long-Short Term Memory (LSTM) models and CNNs (Rei and Yannakoudakis, 2016; Yannakoudakis et al., 2017)

It	changed	my	idea	of	that	classic	music	is	bored	.
C	C	C	I	I	C	I	C	C	I	C

- Exploiting transformer-based language representations such as BERT, GPT-2, etc. (Alikaniotis and Raheja, 2019; Li et al., 2020; Yin et al., 2020; Kaneko et al., 2020; Zhang et al., 2020)
- Predicting edit operations (Awasthi et al., 2019; Omelianchuk et al., 2020; Stahlberg and Kumar, 2020)

^{0} He ^{1} still ^{2} dream ^{3} to ^{4} become ^{5} a ^{6} super ^{7} hero ^{8} . ^{9}

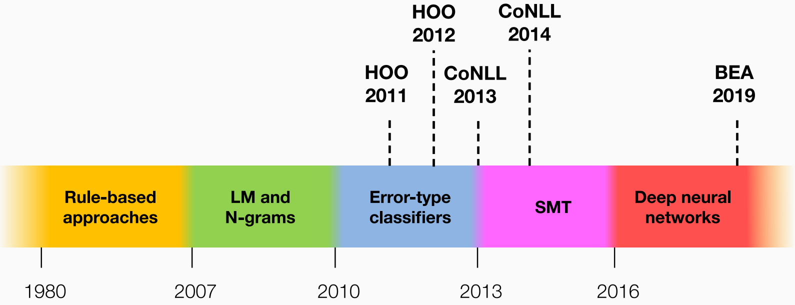
(SELF,2,SELF), (SVA,3,'dreams'), (PART,4,'of'), (FORM,5,'becoming'), (SELF,9,SELF)

Deep neural networks

Strengths and weaknesses

- 👍 Corrects all error types simultaneously
- 👍 End-to-end learning
- 👍 Fluent output
- 👍 State of the art
- 👎 Require lots of parallel training data
- 👎 Very computationally expensive
- 👎 Models are hard to interpret and tweak

Shared tasks



Shared tasks

Shared task	Error types	Corpora	Evaluation	Participants	Approaches	Highest score
HOO 2011 (Dale and Kilgarriff, 2011)	All	Fragments from scientific papers	F_1 for detection, recognition and correction	6		$F_1 = 21.10$
HOO 2012 (Dale et al., 2012)	Det, Prep	FCE (essays from intermediate-level test takers, all backgrounds)	F_1 for detection, recognition and correction	13		$F_1 = 28.70$
CoNLL 2013 (Ng et al., 2013)	Det, Prep, NN, SVA, Vform	NUCLE (essays from Asian backgrounds)	M^2 Scorer (F_1 for correction)	17		$F_1 = 31.20$
CoNLL 2014 (Ng et al., 2014)	All	NUCLE	M^2 Scorer ($F_{0.5}$ for correction)	13		$F_{0.5} = 37.33$
BEA 2019 (Bryant et al., 2019)	All	W&I (essays from all levels and backgrds.) + LOCNESS (essays by native speakers)	ERRANT ($F_{0.5}$ for correction)	21 (restricted) 7 (unrestricted) 9 (low resource)		$F_{0.5} = 69.47$ (restricted) $F_{0.5} = 66.78$ (unrestricted) $F_{0.5} = 64.24$ (low resource)

● Rules ● Language models ● Classifiers ● SMT ● Deep learning

Shared tasks

Other relevant shared tasks

- **Automatic Post-Editing for MT**: correct the output of machine translation systems (ongoing since 2015 as part of the WMT workshop).
- **Automated Evaluation of Scientific Writing**: binary classification of sentences that need grammatical or stylistic correction (Daudaravicius et al., 2016).
- **Chinese Grammatical Error Diagnosis**: identify grammatical errors and their types in non-native Chinese (ongoing since 2014 as part of the NLP-TEA workshop).
- **NLPCC 2018 Shared Task on GEC for Chinese**: correct grammatical errors made by CSL learners (Zhao et al., 2018).
- **QALB Shared Tasks on Automatic Text Correction for Arabic**: correct grammatical errors in native and non-native texts (Mohit et al., 2014; Rozovskaya et al., 2015).

Part III. Data and evaluation

Data and evaluation

Section overview

1. Data annotation
    - Annotation guidelines
    - Preprocessing challenges
2. Corpora
    - Size, domain, annotations
    - Error type frameworks
    - Artificial data
3. Evaluation metrics
    - Strengths and weaknesses
    - Human evaluation

Data annotation

Annotation goals:

- To build a corpus of learner errors
- To let us analyse error patterns
- Training/test data for machine learning

Sample annotation

Dear Paul
I haven't written to you for ages ~~but~~because I was very busy ~~because of~~with ~~the~~ exams at ~~the~~ University. What about you? What's new in ~~Brazil?As~~Brazil? As you know, my friend John asked me to help him with the organization ~~at~~of the concert~~,~~ which was ~~performed~~held last month.

Annotation challenges

Minimal vs. fluent

Original:	*I want explain to you some interesting part from my experience.*
Minimal:	*I want **to** explain to you some interesting **parts of** my experience.*
Fluent:	*I want **to tell you about** some interesting **parts of** my experience.*

Uncorrectable

Original: *She is of the ones that trend to make something enforcing .*

Consistency

- has eating → have eaten
- has → have + eating → eaten

Annotation challenges

Alternative answers

Original	*Social media **has been playing a vital important** role in our lives today .*
A1	*Social media **plays an important** role in our lives today .*
A2	*Social media **plays a vital** role in our lives today .*
A3	*Social media **play a vitally important** role in our lives today .*
A4	*Social media **plays a vital** role in our lives today .*
A5	*Social media **plays a vital and important** role in our lives today .*
A6	*Social media **plays a vitally important** role in our lives today .*
A7	*Social media **has been playing a vital important** role in our lives today .*
A8	*Social media **plays a vital , important** role in our lives today .*
A9	*Social media **is playing a vital important** role in our lives today .*
A10	*Social media **has been playing a vital** role in our lives today .*

Corpus processing challenges

Whitespace anomalies

- *Let's discuss ~~about~~ this* → *Let's discuss␣␣this*

Fluid sentence boundaries

- *I liked it~~.~~, but he didn't~~. So~~, so we left.*

Character-to-token edits

Token	Edit	Problem
WORD.	. → ,	Tokeniser
dancing	ing → ed	Guidelines
To	T → to	Carelessness

→ Annotator guidelines are very important!

Corpora: FCE

Name	First Certificate in English
Train	28k sentences, 454k tokens
Dev	2.2k sentences, 35k tokens
Test	2.7k sentences, 42k tokens
Level	Intermediate (B1-B2)
Edits	Yes (77 types)
Domain	Short essays, letters, exams
Authors	International ESL learners
Notes	One of the earliest public corpora (2011); Official corpus of the HOO-2012 shared task; Can also be used for other tasks; e.g. essay scoring; A subset of the Cambridge Learner Corpus;
Reference	Yannakoudakis et al. (2011)

Corpora: CLC

Name	Cambridge Learner Corpus
Train	2m sentences, 29m tokens
Dev	-
Test	-
Level	Beginner - Advanced (A1-C2)
Edits	Yes (77 types)
Domain	Short essays, letters, exams
Authors	International ESL learners
Notes	Largest, professionally annotated corpus; Annotated since 1993; Private, commercial corpus; Can also be used for other tasks; e.g. essay scoring;
Reference	Nicholls (2003)

Corpora: NUCLE

Name	National University of Singapore Corpus of Learner English
Train	57k sentences, 1.1m tokens
Dev	-
Test	-
Level	Upper Intermediate (C1)
Edits	Yes (28 types)
Domain	Essays
Authors	South-East Asian Undergraduates
Notes	The first purpose-built GEC corpus; Official training corpus of CoNLL-2013/2014; Only 40% of sentences contain errors; A bit noisy; URLs and bibliographies;
Reference	Dahlmeier et al. (2013)

Corpora: CoNLL-2013/2014

Name	Conference on Natural Language Learning shared tasks
Train	-
Dev	1.4k sentences (29k tokens) – CoNLL-2013
Test	1.3k sentences (30k tokens) – CoNLL-2014
Level	Upper Intermediate (C1)
Edits	Yes (28 types)
Domain	Essays
Authors	South-East Asian Undergraduates
Notes	CoNLL-2013 was originally a test set; CoNLL-2014 has 10 references (2 official, 8 extended); CoNLL-2014 is still a common benchmark; Very narrow domains: i) technology, ii) genetic testing;
Reference	Ng et al. (2013, 2014)

Corpora: WikEd

Name	WikEd Error Corpus
Train	12.1m sentences, 292m tokens
Dev	-
Test	-
Level	Native
Edits	Yes (untyped)
Domain	Wikipedia articles
Authors	Native speakers
Notes	One of the largest corpora with edits; Extracted from Wikipedia revision history; Wikipedia revisions are not always grammatical edits; A preprocessed version is available (4.7m sentences);
Reference	Grundkiewicz and Junczys-Dowmunt (2014)

Corpora: Lang-8

Name	Lang-8 Corpus of Learner English
Train	1m sentences (11.8m tokens)
Dev	-
Test	-
Level	Unclear; Beginner - Advanced (A1-C2)?
Edits	No
Domain	Web
Authors	International - many Japanese L1
Notes	One of the largest public corpora; Noisy – not professionally annotated; A cleaned subset of the multilingual Lang-8 Learner Corpus;
Reference	Mizumoto et al. (2011); Tajiri et al. (2012)

Corpora: JFLEG

Name	Johns Hopkins Fluency-Extended GUG Corpus
Train	-
Dev	754 sentences (14k tokens)
Test	747 sentences (14k tokens)
Level	Unknown
Edits	No
Domain	Essays
Authors	ESL learners
Notes	Advocated fluent over minimal corrections; 4 sets of references (both dev and test); Isolated sentences (not whole essays); Smallest test set;
Reference	Napoles et al. (2017)

Corpora: W&I + LOCNESS

Name	Cambridge English Write & Improve and LOCNESS
Train	34k sentences (628k tokens)
Dev	4.4k sentences (87k tokens)
Test	4.5k sentences (86k tokens)
Level	Beginner - Advanced (A1-C2), Native (LOCNESS)
Edits	Yes (55 types - automatic)
Domain	Short essays, letters, exams, web
Authors	International ESL learners
Notes	Native LOCNESS data only in dev and test;
	Balanced across all ability levels in terms of sentences;
	Released with the BEA-2019 shared task;
	Official dev/test data of the BEA-2019 shared task;
	5 sets of references in the test data;
Reference	Bryant et al. (2019)

Frameworks: CLC

Error types: 77

- 8 prefix operation/morphology codes

M	missing	R	replacement	U	unnecessary	AG	agreement
C	countability	D	derivation	F	form	I	inflection

- 10 suffix POS codes

A	pronouns	C	conjunctions	D	determiners	J	adjectives
N	nouns	P	punctuation	Q	quantifiers	T	prepositions
V	verbs	Y	adverbs				

- 12 separate codes

AS	arg. structure	CE	compounds	CE	collocations	ID	idioms
L	register	QL	prompt error	S	non-word sp.	SA	US spelling
SX	real word sp.	TV	verb tense	W	word order	X	negation

See Nicholls (2003) for more information

Frameworks: CLC

Strengths and weaknesses

- 👍 Very detailed types
- 👍 Modular system
- 👍 Easy to extract error patterns based on types
- 👎 Complex; annotators need extensive training
- 👎 Sparse; 50/77 types each account for <1% of all edits

Frameworks: NUCLE

Error types: 28

Vt	Verb tense	Wtone	Tone (formal/informal)
Vm	Verb modal	Srun	Run-on sentence, comma splice
V0	Missing verb	Smod	Dangling modifiers
Vform	Verb form	Spar	Parallelism
SVA	Subject-verb agreement	Sfrag	Sentence fragment
ArtOrDet	Article or determiner	Ssub	Subordinate clause
Nn	Noun number	WOinc	Word order
Npos	Noun possessive	WOadv	Adjective/adverb order
Pform	Pronoun form	Trans	Conjunctions/linking words
Pref	Pronoun reference	Mec	Spelling, punctuation, etc.
Prep	Preposition	Rloc-	Redundancy
Wci	Wrong collocation/idion	Cit	Citation
Wa	Acronym	Others	Other errors
Wform	Word form	Um	Unclear meaning

See Dahlmeier et al. (2013) for more information

Frameworks: NUCLE

Strengths and weaknesses

- 👍 Much smaller than the CLC framework
- 👍 Only 9/28 types each account for < 1% of all edits
- 👍 Syntactic error types; e.g. parallelisms, sentence fragments
- 👎 Not modular; many types have inconsistent scope
  - Vform vs. Wform, WOadv vs. WOinc
  - Rloc- vs. ArtOrDet/Prep
- 👎 Some extremely specific types
  - Citations (Cit)
  - Acronyms (Wa)

Frameworks: ERRANT

Error types: 55

- 3 prefix operation codes

M	Missing	R	Replacement	U	Unnecessary

- 25 main codes

POS		**Morphology**		**Other**	
ADJ	Adjective	ADJ:FORM	Adjective form	CONTR	Contractions
ADV	Adverb	NOUN: INFL	Noun inflection	ORTH	Orthography
CONJ	Conjunction	NOUN:NUM	Noun number	OTHER	Other
DET	Determiner	NOUN:POSS	Noun possessive	SPELL	Spelling
NOUN	Noun	VERB:FORM	Verb form	UNK	Unknown
PART	Particle	VERB:INFL	Verb inflection	WO	Word order
PREP	Preposition	VERB:SVA	Subject-verb agreement		
PRON	Pronoun	VERB:TENSE	Verb tense		
PUNCT	Punctuation	MORPH	Other morphology		
VERB	Verb				

See Bryant et al. (2017) for more information

Frameworks: ERRANT

Strengths and weaknesses

- 👍 Fully automatic annotation
- 👍 Immune to annotator bias
- 👍 Modular system (inspired by CLC)
- 👍 Interpretable; type reasoning recoverable from rules
- 👍 30/55 categories each account for < 1% of all edits
- 👎 Longer multi-token edits often classified as OTHER
- 👎 Dependent on other resources (spaCy, word list)

Artificial data

Use artificial data to support limited manual data.

Generation methods:

- Applying edits from real data to a native corpus.
- Matching error type distributions of real data.
- Generating from spellcheckers.
- Noisy back-translation.

Active area of research:

- Artificial data quality can have a big impact.

Corpus recommendations (opinion)

Training (ordered by priority)

- W&I+LOCNESS, FCE, Lang-8, NUCLE
- + Artificial data?

Development

- General purpose: W&I+LOCNESS
- In-domain dataset

Testing

- W&I+LOCNESS (largest, most balanced, recent)
- CoNLL-2014, FCE (compare with previous work)
- JFLEG (smallest, perhaps less informative)

Evaluation

Most commonly carried out in terms of edits

Original	I often look at TV	Span-based	Span-based	Token-based
Reference	[2, 4, watch]	Correction	Detection	Detection
Hypothesis 1	[2, 4, watch]	Match	Match	Match
Hypothesis 2	[2, 4, see]	No match	Match	Match
Hypothesis 3	[2, 3, watch]	No match	No match	Match

Problem: unannotated hypothesis vs. annotated reference

Original	*This is grammatical sentences .*
Hypothesis	*This **are** **a** grammatical **sentences** .*
Reference	*This is a grammatical sentence .*
Gold edits	[2, 2, a], [3, 4, sentence]

Metrics: HOO Scorer

Reference: Dale and Kilgarriff (2011)

Motivation: the HOO-2011/12 shared tasks

Intuition:

1. Align the original and hypothesis using Levenshtein
2. Compare the hypothesis edits to the reference edits
3. Use TP, FP, FN to compute F-score

Correction (span-based correction);
Recognition (span-based detection);
Detection (token-based detection);

No longer used, but inspired subsequent metrics.

Metrics: HOO Scorer

Strengths and weaknesses

- 👍 Simple and intuitive
- 👍 Interpretable
- 👍 Detection and correction scores
- 👎 Automatic alignment may not match human alignment
  - has eat → have eaten vs. has → have + eat → eaten
- 👎 Unchanged words in edits are never matched
  - house → the house

Metrics: MaxMatch (M^2) Scorer

Reference: Dahlmeier and Ng (2012b)

Motivation: weaknesses in the HOO scorer

Intuition:

1. Align the original and hypothesis using Levenshtein
2. Dynamically choose the alignment that maximally matches the reference edits
3. Use TP, FP, FN to compute F-score

Official scorer of the CoNLL-2013/14 shared tasks.
Since CoNLL-2014, we use $F_{0.5}$:

- $F_{0.5}$ weights Precision twice as much as Recall

Still used today, notably on the CoNLL-2014 test set.

Metrics: MaxMatch (M^2) Scorer

Strengths and weaknesses

- 👍 Dynamic edit spans
- 👍 Interpretable
- 👎 Cannot discriminate between a do-nothing baseline and a system that only proposes bad corrections
- 👎 Partial matches are ignored
  - Hyp: eat → eaten vs. Ref: is eat → has eaten
- 👎 False positive (FP) count is artificially reduced

Original:	*He* ***looked*** *at* ***the*** *cat .*
Hypothesis:	*He* ***looks*** *at* ***a*** *cat .*
M^2 Edit:	looked at the → looks at a = 1FP
Human Edit:	looked → looks, the → a = 2FP

Metrics: *I*-measure

Reference: Felice and Briscoe (2015)

Motivation: weaknesses in the M^2 scorer

Intuition:

1. Carry out a 3-way alignment of orig, hyp and ref
2. Classify each token (not span) as a TP, TN, FP, FN
3. Compute (weighted) accuracy for the system: WAcc_{sys}
4. Do the same for a do-nothing baseline (hyp = orig): $\text{WAcc}_{\text{base}}$
5. Compare WAcc_{sys} with $\text{WAcc}_{\text{base}}$ to compute *Improvement*

Improvement (*I*) may be positive or negative

Metrics: *I*-measure

Strengths and weaknesses

- 👍 Discriminates between bad systems and do-nothing systems
- 👍 Rewards partial matches
- 👍 Interpretable for both improvement and degradation
- 👎 Does not correlate with human judgements
- 👎 Completely reordered the CoNLL-2014 rank results
- 👎 The value of the weight in weighted accuracy is arbitrary

I-measure rarely used in practice.

Metrics: GLEU

Reference: Napoles et al. (2015)

Motivation: overcome the dependency on edits

Intuition:

- Inspired by BLEU n-gram matching
- Reward hyp n-grams that match ref, but not orig
- Penalise hyp n-grams that match orig, but not ref
- Average scores over different references

Developed for fluency and JFLEG;
GLEU+ Napoles et al. (2016) removed a tunable weight;
Not to be confused with Google BLEU (GLEU) (Wu et al., 2016);
Often only reported on JFLEG.

Metrics: GLEU

Strengths and weaknesses

- 👍 Requires parallel sentences rather than reference edits
- 👍 Claimed to correlate more strongly with human judgements
- 👎 Uninterpretable: higher doesn't necessarily mean better
- 👎 Non-deterministic due to reference averaging
- 👎 Strongly correlates with recall
- 👎 Low discriminative power: e.g. 68-78 GLEU ≈ 40-75 $F_{0.5}$

Metrics: ERRANT

Reference: Bryant et al. (2017)

Motivation: facilitate error type scores

Intuition:

- Align orig and hyp using custom, linguistically-enhanced Damerau-Levenshtein (POS, lemma, chars)
- Use rules to merge parts of the alignment
- Use rules to automatically classify hyp edits
- Use TP, FP, FN to compute overall and error type F-scores

A variant of the HOO/M^2 scorer
Official scorer of the BEA-2019 shared task
Can also be used to standardise corpus annotation

Metrics: ERRANT

Strengths and weaknesses

- 👍 Automatic annotation
- 👍 Facilitates detailed error type analysis
- 👍 Detection and correction scores
- 👍 Interpretable
- 👎 Dependent on other resources (spaCy, word list)
- 👎 Cannot discriminate between a do-nothing baseline and a system that only proposes bad corrections

Mismatch: Auto hyp edits vs. gold ref edits
Solution: Convert gold ref edits to auto ref edits

Metrics: Others

Ratio Scoring (Bryant and Ng, 2015)

- Motivation: Humans vs. humans do not reach 100 $F_{0.5}$
- Get the average $F_{0.5}$ of each annotator vs. other annotators
- Ratio score = system $F_{0.5}$ / average human $F_{0.5}$

USim (Choshen and Abend, 2018b)

- Motivation: No metric incorporates semantic similarity
- Semantically parse orig+hyp and orig+ref and compare trees

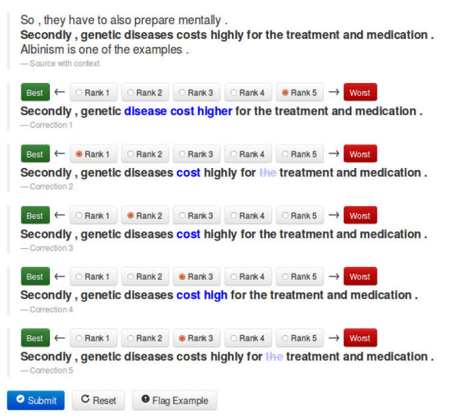
Syntatic Errors and Classification (SErCl) (Choshen et al., 2020)

- A variant of ERRANT that only uses Universal Dependencies
- Multilingual, although with limitations

Human evaluation

Experiments inspired by the WMT human evaluation campaign

- Humans rank different subsets of corrected sentences
- This is used to infer an overall human ranking
- Correlate human ranking vs. metric ranking of different systems



Human evaluation

Previous work: Rank 12 CoNLL 2014 systems + source at the corpus-level

- Grundkiewicz et al. (2015) (8 raters); Napoles et al. (2015) (3 raters)

	Napoles et al.		Grundkiewicz et al.	
Metric	**r**	**ρ**	**r**	**ρ**
M^2 $F_{0.5}$	0.358	0.429	0.692	0.629
M^2 $F_{0.18}$	-	-	0.758	0.701
I-measure	-0.051	-0.005	-0.154	-0.098
GLEU	0.542	0.555	-	-
BLEU	-0.125	-0.225	-0.346	-0.24

- Chollampatt and Ng (2018b) added statistical significance tests and sentence correlation
- Choshen and Abend (2018a) hypothesised large variance because inter-rater agreement is low

Human evaluation

Latest work: Napoles et al. (2019) (8 raters)

- 1000 sentences from 3 different datasets
- All sentences were judged to avoid sampling bias

	FCE		WikEd		Yahoo	
Metric	r	ρ	r	ρ	r	ρ
M^2 $F_{0.5}$	0.860	0.849	0.346	0.552	0.580	0.699
I-measure	0.819	0.839	0.854	0.875	0.915	0.900
GLEU	0.838	0.813	0.426	0.538	0.740	0.775
ERRANT $F_{0.5}$	0.919	0.887	0.401	0.555	0.532	0.601
GMEG-Metric	0.984	0.950	0.982	0.967	0.940	0.931
Human	0.992	0.931	0.994	0.907	0.988	0.990

- It's problematic if correlation depends on dataset
- GMEG-Metric is consistent but …
  … it's trained on 73 features from 6 different metrics.

Human evaluation

Future work

- Human evaluation in GEC is an unsolved problem
- Experiments have been done on a small scale
- Sentence-based ratings are problematic

Original	*Social media **has been playing a vital important** role in our lives today .*
A1	*Social media **plays an important** role in our lives today .*
A2	*Social media **plays a vital** role in our lives today .*
A3	*Social media **play a vitally important** role in our lives today .*

- Intuitively, some errors are more serious than others
  - e.g. M:DET vs. M:VERB
- More research needed

Metric recommendations (opinion)

Current trends

- M^2 $F_{0.5}$ (CoNLL-2014)
- GLEU (JFLEG)
- ERRANT $F_{0.5}$ (BEA-2019)

  … but it seems unwise to use 3 different metrics for 3 different datasets.

Recommendations (ordered by priority)

1. ERRANT $F_{0.5}$
   - Only ERRANT can provide detailed feedback for detection, correction and error types (English only)
2. M^2 $F_{0.5}$
   - Mainly useful for comparison with previous work
3. GLEU
   - Mainly used with JFLEG, but JFLEG is very small

Part IV. Neural grammatical error correction

Neural GEC

Section overview

1. Neural approach to GEC
2. GEC as a low-resource NMT task
3. Data sparsity
4. Correction efficacy
5. Beyond the NMT framework

Neural GEC

GEC as (neural) machine translation

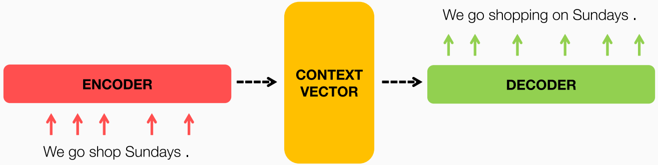
“Incorrect” English $\rightarrow$ “Correct” English

A large number of well-established methods from NMT can be applied to and adapted for GEC.

Terminology: translation = correction, source text = erroneous input text, target text = corrected output text, back-translation $\approx$ error generation

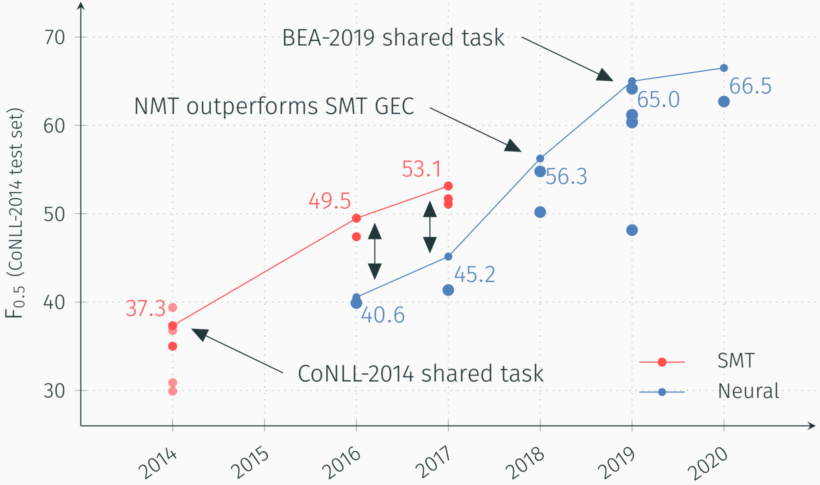
The encoder-decoder architecture

Refer to Koehn (2020); Stahlberg (2020) or other for more details.



- Training on parallel sentence pairs using a gradient-based optimizer and cross-entropy loss; decoding with beam search.
- Recurrent Neural Networks (RNN) (Bahdanau et al., 2015; Miceli Barone et al., 2017), Convolutional Neural Networks (CNN) (Gehring et al., 2017), Transformer (Vaswani et al., 2017).

Progress in GEC on CoNLL-2014



GEC as low-resource NMT

1. “A Multilayer Convolutional Encoder-Decoder Neural Network for Grammatical Error Correction”, Chollampatt & Ng, AAAI 2018
2. “Approaching Neural Grammatical Error Correction as a Low-Resource Machine Translation Task”, Junczys-Dowmunt et al., NAACL 2018

Subword segmentation, domain adaptation, strong regularization with dropout, transfer learning, model ensembles, utilizing a language model, deeper models, and others.

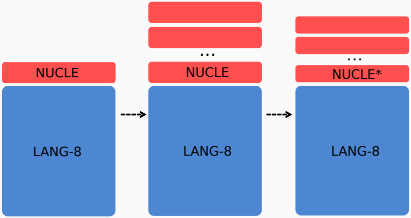
GEC as low-resource NMT

Subword segmentation

It	was	really	exiting	and	unforgetable	experience	.
					↓		
It	was	really	exiting	and	un@@ forget@@ able	experience	.

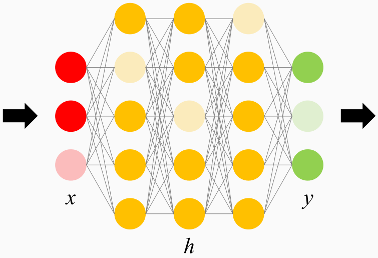
- Rare words are split into frequent sub-word units using the byte pair encoding (BPE) algorithm (Sennrich et al., 2016).
- Early neural GEC systems restore unknown words via word alignments (Yuan et al., 2016) or operate at the character level (Xie et al., 2018).

GEC as low-resource NMT



- **Domain adaptation** by oversampling the in-domain NUCLE corpus 10 times.
  - NUCLE: 57.1K sentences, Lang-8: 1.2M
- **Error rate adaptation** by removing random clean sentence pairs from the oversampled NUCLE data.
  - NUCLE: 6% WER, CoNLL-2013: 15% WER.

GEC as low-resource NMT



- Strong regularization with **dropout** (Srivastava et al., 2014).
- Dropout over source words as a noising strategy.
  - The full embedding vector is set to 0 with a probability p_{src}, all other embedding values are scaled with $1/(1-p_{\mathrm{src}})$.

GEC as low-resource NMT

Transfer learning

- Pre-training parts of the neural network on another task using monolingual data.
  - Initializing embedding vectors with pre-trained word embeddings (e.g. *word2vec*, *GloVe*, *fastText*).
  - Initializing the decoder parameters with a pre-trained language model.

$\rightarrow$ Pre-training with the denoising autoencoder.

Model ensembles

- Ensemble of independently trained models.
  - Predictions from each of the individual models are averaged to improve the performance.
  - Ensembling weak models may lower M^2 score due to precision bias.
  - Computationally expensive.
- Combining with a language model.
  - Weighted ensemble with LM (weights are optimized on a development set).
  - Rescoring the list of n-best correction candidates.
- Single models: averaging model checkpoints or exponential smoothing of model parameters.

Further research
on neural GEC

↙ ↓ ↘

Overcoming data sparsity

Improving correction efficacy

Beyond the NMT framework

Data sparsity

Pre-training word embeddings or decoder parameters on **clean monolingual texts** → Pre-training the entire encoder-decoder jointly on generated **artificial parallel data**

* Good quality artificial data can be used to augment the human-annotated training data.

Data sparsity

Approaches to artificial error generation:

A. Random perturbations to clean monolingual texts (unsupervised).
B. Error generation based on the error distributions of annotated corpora.
C. Using other parallel corpora, e.g. Wikipedia revisions, machine translation corpora.
D. In-training generation of additional error examples.

A. Random perturbations

- Corrupting clean sentences by random substitution, deletion, insertion or reordering of words/characters with a small probability (Xie et al., 2018; Zhao et al., 2019).
- An unsupervised approach to error generation, e.g. for denoising autoencoders.
- Restricting word substitutions (the most frequent edit operation) to *confusion sets* of possible error patterns generated with a spell-checker (Grundkiewicz et al., 2019).
- The generated noise is not always *grammatical* errors.

Artificial error synthesis with spell-checking

(Grundkiewicz et al., 2019)

Aspell dict.

→

Aspell's suggestion lists:

had	hard head hand gad has ha ad hat ...
night	knight naught nought nights bight might ...
then	them the hen ten than thin thee thew ...

+

News Crawl 100M

+

Random substitution /deletion /insertion of words

Orig. *The ideal ratio is to spend no more than about 30 percent of a salary on housing .*

+ Synth. *The ideal ratios to is spend no more than ~~about~~ 30 percent of a salary on housing .*

+ Spell. *The ideal ratios to is spend no more than 30 percent of a slary on housing .*

B. Artificial error generation from a seed corpus

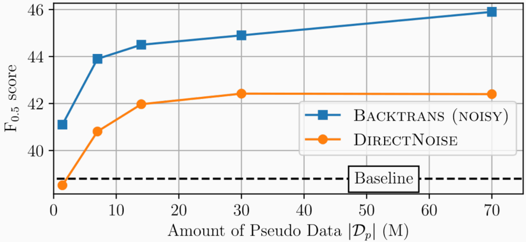
- Extracting error patterns (edits) and conditional probabilities from an error corpus and applying them to clean text with linguistically-motivated heuristic rules (e.g. Felice et al. (2014)).
    - Token-based noising via error patterns and type-based noising for prepositions, nouns, and verbs (Choe et al., 2019).
- A machine translation system/sentence transduction model trained on pairs of (*corrected*, *erroneous*) sentences (Rei et al., 2017; Kasewa et al., 2018; Kiyono et al., 2019).
    - Neural methods benefit from noising strategies during decoding, e.g. random sampling or temperature sampling.

“Back-translation” vs noising

(Kiyono et al., 2019)

Backtrans (Noisy) Penalising hypotheses in the beam by adding $r\beta_{random}$ to the score at every time step.

DirectNoise Adding noise to the training sentence by deletion, insertion or masking a token.

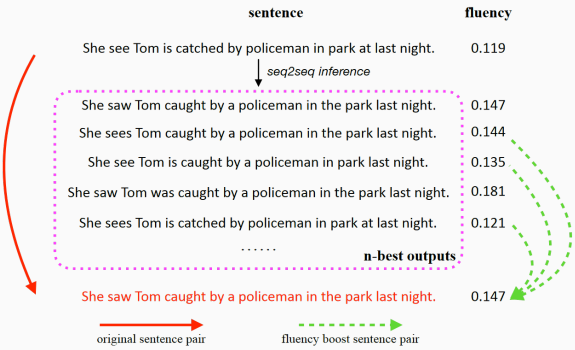


Graph source: Kiyono et al. (2019)

C. Utilizing other parallel corpora

- Building an error corpus from Wikipedia revision histories (*WikEd*, Grundkiewicz and Junczys-Dowmunt (2014)).
  - Large, but noisy and represents a different domain.
- Round-trip translation via a bridge language, e.g. translating from English to Chinese to English (Lichtarge et al., 2019).
- Different quality of MT systems can resemble different proficiency levels of writers (Zhou et al., 2020).
  - Translation from a weak MT system $\rightarrow$ source sentence; translation from a strong MT system $\rightarrow$ target sentence.

D. In-training error generation



- Generating additional training examples from the n-best outputs during training, e.g. **fluency boost learning** (Ge et al., 2018a).
- Orthogonal to other methods.

Example source: Ge et al. (2018a)

Which method for artificial error generation is best?

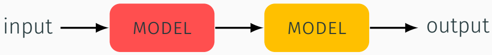
- Matching the error distribution of the targeted domain improves performance on domain-specific testsets (cf. White and Rozovskaya (2020)).
- Context-aware error generation may scale better (Kiyono et al., 2019).
- An unsupervised method can be used for other languages or in very low-resource scenarios (Náplava and Straka, 2019; Grundkiewicz and Junczys-Dowmunt, 2019).

How to utilize the artificial training data for best performance?

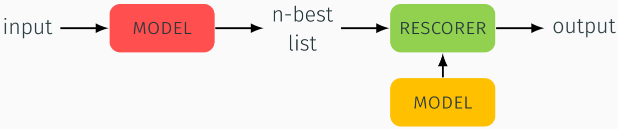
- Top NMT-based systems use up to 100M artificial parallel sentences.
- With a sufficient amount of artificial data, pre-training and fine-tuning works better than augmenting the original training data (Kiyono et al., 2019).
- Fine-tuning on a dataset combining the original and artificial training data can be more effective than fine-tuning on original data only (Grundkiewicz et al., 2019; Omelianchuk et al., 2020).

Correction efficacy

Output pipelining:

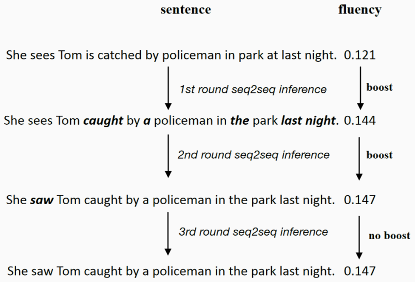


Rescoring n-best outputs:



- Incremental (iterative) correction.
- Combining left-to-right and right-to-left models.
- Handling spelling errors.

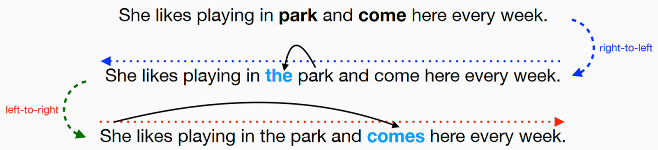
Incremental correction



- Incremental correction with a single system.
  - **Fluency boost inference** rewrites the output only if the *fluency score* increases (Ge et al., 2018a).
  - **Iterative decoding** processes the 2nd best output if the best output contained no edits (Lichtarge et al., 2019).
- Works best for a high-precision system.
- Computationally expensive.

Figure source: Ge et al. (2018a)

Right-to-left models



- Motivation: some error types are easier to correct in the right-to-left direction (e.g. articles), others in the left-to-right direction (e.g. subject-verb agreement).
- A right-to-left model is trained on the reversed token order.
    - System pipelining (Ge et al., 2018b) or re-ranking the n-best list by right-to-left models (Grundkiewicz et al., 2019).

Figure source: Ge et al. (2018b)

Spell-checking

Spelling errors often form out-of-vocabulary words, which pose a challenge to word-level sequence-to-sequence models.

- Data pre-processing with a traditional spell-checker (e.g. JFLEG).
- Contextual spell-checking in post-processing (Chollampatt and Ng, 2018a; Choe et al., 2019).
- Random character-level perturbations in the source sentences in the training data (Lichtarge et al., 2019; Junczys-Dowmunt et al., 2018).

Precision vs recall

$\longleftarrow$ higher precision vs higher recall $\longrightarrow$

Decreasing WER	Increasing WER Edit-weighted MLE
Model ensembling	Ensembling with LM
N-best list rescoring	System pipelines Incremental decoding
…	…

Beyond NMT

Criticism of the NMT-based approach to GEC

- 👎 NMT architectures are tailored to bilingual tasks.
  - Grammatical error correction is a **monolingual task**.
  - Most tokens are copied, which seems wasteful.
- 👎 Slow inference, especially with methods like incremental decoding, ensembling or rescoring.
- 👎 Difficult interpretability and explainability.

Beyond NMT

GEC-specific adaptations of the neural-based approach

- Word-level edit operations (word insertions, deletions, substitutions).
  - Rescoring with edit operation features (Chollampatt and Ng, 2018a).
  - Edit-weighted training objective (Junczys-Dowmunt et al., 2018).
- Copy-augmented architectures.
  - A copying mechanism adds the ability to copy tokens from the input sequence to the output (Zhao et al., 2019).
- Using large pre-trained contextual language models like BERT (Devlin et al., 2019).
  - Best performance using features from BERT fine-tuned with GEC data (Kaneko et al., 2020).

Beyond NMT

Sequence editing models

fowler fed dog. $\rightarrow$ Fowler fed the dog. vs

fowler fed dog. $\rightarrow$ `capitalize 1`, `append(the) to 2`, `copy 3`

- Sequence tagging instead of sequence generation (Awasthi et al., 2019; Omelianchuk et al., 2020; Stahlberg and Kumar, 2020).
    - Utilizing pre-trained models like BERT, XLNet, RoBERTa.
    - Faster than the traditional NMT approach;
      some models use non-auto-regressive decoding.
- Promising research direction, but an exhaustive comparison with the NMT approach is needed.

Example from Awasthi et al. (2019)

Summary

Which methods are best?

- Basic methods for each GEC system
  (e.g. subword segmentation, dropout, domain adaptation, etc.).
- Use tools according to the needs
  (e.g. precision vs recall, very low-resource scenarios).

- Check error-annotated data used for training.
- Re-evaluate methods developed for shared tasks.
- Build own baselines.

Part V. Recent and future work

Section overview

1. Findings of the BEA-2019 shared task
2. Towards unsupervised GEC
3. Non-English GEC
4. Where next?

BEA-2019 shared task

Task overview

- 3 tracks: Restricted, Unrestricted, Low Resource
- Restricted data: FCE, NUCLE, Lang-8, W&I+LOCNESS
- 24 unique teams took part (21 in Restricted)
- Two-thirds of all teams use Transformer NMT (most of the remainder used CNNs)
- Evaluation in terms of ERRANT $F_{0.5}$
- Reference: Bryant et al. (2019)

https://www.cl.cam.ac.uk/research/nl/bea2019st/

BEA-2019 shared task

Findings

- Most systems performed best on missing word errors
- Content word errors (ADJ, ADV, CONJ, NOUN, OTHER, VERB) were amongst the hardest to correct (< 50 $F_{0.5}$)
- There is room for improvement on multi-token errors
- $\sim$15 $F_{0.5}$ difference between correction and token detection

https://www.cl.cam.ac.uk/research/nl/bea2019st/

BEA-2019 shared task: Practical information

Evaluation via Codalab remains open to anonymous submissions

- This is the only way to evaluate on BEA-test
- References are currently withheld to ensure fairness

All participating system output is publicly available.

Be wary of comparing CoNLL-2014 and BEA-test scores:

- 65 $F_{0.5}$ M^2 (CoNLL-2014) $\not\approx$ 70 $F_{0.5}$ ERRANT (BEA-test)
- M^2 slightly inflates scores due to FP merging
- 2 references (CoNLL-2014) vs 5 references (BEA-test)
- Easier to overfit to narrow CoNLL-2014 domain

https://www.cl.cam.ac.uk/research/nl/bea2019st/

Towards unsupervised GEC

Language models:

- Define or generate a confusion set for a given token and score alternatives using a language model.
- Use a small development set to tune score thresholds.
- Recent work has used a large 5-gram model or transformer masked language model (Bryant, 2018; Alikaniotis and Raheja, 2019; Stahlberg et al., 2019; Sun and Jiang, 2019).

Unsupervised error generation:

- Pre-training on artificial parallel data generated with the inverted spell-checker method + fine-tuning on a development set if available.
- State-of-the-art results for Czech, German and Russian in low-resource scenarios (Náplava and Straka, 2019; Grundkiewicz and Junczys-Dowmunt, 2019).

Non-English languages

Publicly available error corpora for non-English languages:

Arabic: the QALB corpora (Rozovskaya et al., 2015) includes corrected texts produced by native and non-native writers, as well as MT output.

Chinese: the NLPCC 2018 test set (Zhao et al., 2018) extracted from the *PKU Chinese Learner Corpus* composed of essays written by foreign college students of Mandarin Chinese.

Czech: the AKCES-GEC corpus (Náplava and Straka, 2019) contains manually annotated transcripts of essays of non-native speakers of Czech.

German: the Falko-MERLIN GEC corpus (Boyd, 2018) combines two German learner corpora of all proficiency levels.

Russian: the RULEC-GEC dataset (Alsufieva et al., 2012; Rozovskaya and Roth, 2019) consists of Russian texts from foreign and heritage speakers.

Non-English languages

Lang.	Corpus	Dev	Test	Train
Arabic	QALB-2014 (Mohit et al., 2014)	1,017	968	20,428
	QALB-2015 (Rozovskaya et al., 2015)	25K words	23K words	43K words
Chinese	NLPCC 2018 (Zhao et al., 2018)	–	2,000	–
	Lang-8	–	–	717,241
Czech	AKCES-GEC (Náplava and Straka, 2019)	2,485	2,676	42,210
German	Falko-MERLIN GEC (Boyd, 2018)	2,503	2,337	20,237
	+ Wikipedia edits	–	–	1M+
Russian	RULEC-GEC (Rozovskaya and Roth, 2019)	2,500	5,000	4,980
English	W&I+LOCNESS (Bryant et al., 2019)	4,384	4,477	34,308

Corpus sizes in the number of source sentences except for QALB-2015.

Future work

Better resources

- Error corpora for non-English languages

Better systems

- Move beyond sentence-based GEC (Chollampatt et al., 2019)
- Semantic errors
- Personalised GEC (e.g. L1 and ability level)
- Unsupervised/Low resource approaches

Better evaluation

- More robust human evaluation
- Better automatic evaluation metrics

Thanks for (virtually) attending this tutorial!

We look forward to any questions :)

Roman Grundkiewicz, Christopher Bryant, Mariano Felice
rgrundki@inf.ed.ac.uk, {cjb255,mf501}@cl.cam.ac.uk

References

Alikaniotis, D. and Raheja, V. (2019). The unreasonable effectiveness of transformer language models in grammatical error correction. In Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, pages 127–133, Florence, Italy. Association for Computational Linguistics.

Alsufieva, A., Kisselev, O., and Freels, S. (2012). Results 2012: Using flagship data to develop a Russian learner corpus of academic writing. Russian Language Journal, 62:79–105.

Andersen, Ø. E., Yannakoudakis, H., Barker, F., and Parish, T. (2013). Developing and testing a self-assessment and tutoring system. In Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications, BEA 2013, pages 32–41, Atlanta. Association for Computational Linguistics.

Arppe, A. (2000). Developing a grammar checker for swedish. In Proceedings of the Twelfth Nordic Conference in Computational Linguistics (NoDaLiDa), pages 13––27, Trondheim, Norway.

Awasthi, A., Sarawagi, S., Goyal, R., Ghosh, S., and Piratla, V. (2019). Parallel iterative edit models for local sequence transduction. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 4260–4270, Hong Kong, China. Association for Computational Linguistics.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In ICLR 2015.

Berend, G., Vincze, V., Zarrieß, S., and Farkas, R. (2013). LFG-based features for noun number and article grammatical errors. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task, pages 62–67, Sofia, Bulgaria. Association for Computational Linguistics.

Bergsma, S., Lin, D., and Goebel, R. (2009). Web-scale n-gram models for lexical disambiguation. In Proceedings of the 21st International Joint Conference on Artifical Intelligence, IJCAI'09, pages 1507–1512, Pasadena. Morgan Kaufmann Publishers Inc.

Boroş, T., Dumitrescu, S. D., Zafiu, A., Barbu Mititelu, V., and Vaduva, I. P. (2014). Racai gec – a hybrid approach to grammatical error correction. In Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task, pages 43–48, Baltimore, Maryland. Association for Computational Linguistics.

Boyd, A. (2018). Using Wikipedia edits in low resource grammatical error correction. In Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text, pages 79–84, Brussels, Belgium. Association for Computational Linguistics.

Brockett, C., Dolan, W. B., and Gamon, M. (2006). Correcting ESL Errors Using Phrasal SMT Techniques. In Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, pages 249–256, Sydney, Australia. Association for Computational Linguistics.

Bryant, C., Felice, M., Andersen, Ø. E., and Briscoe, T. (2019). The BEA-2019 shared task on grammatical error correction. In Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, pages 52–75, Florence, Italy. Association for Computational Linguistics.

Bryant, C., Felice, M., and Briscoe, T. (2017). Automatic annotation and evaluation of error types for grammatical error correction. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 793–805, Vancouver, Canada. Association for Computational Linguistics.

Bryant, C. and Ng, H. T. (2015). How far are we from fully automatic high quality grammatical error correction? In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 697–707, Beijing, China. Association for Computational Linguistics.

Bryant, Christopherand Briscoe, T. (2018). Language Model Based Grammatical Error Correction without Annotated Training Data. In Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications, pages 247–253, New Orleans, Louisiana. Association for Computational Linguistics.

Buys, J. and van der Merwe, B. (2013). A tree transducer model for grammatical error correction. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task, pages 43–51, Sofia, Bulgaria. Association for Computational Linguistics.

Cahill, A., Chodorow, M., Wolff, S., and Madnani, N. (2013a). Detecting missing hyphens in learner text. In Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications, pages 300–305, Atlanta, Georgia. Association for Computational Linguistics.

Cahill, A., Madnani, N., Tetreault, J., and Napolitano, D. (2013b). Robust systems for preposition error correction using wikipedia revisions. In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 507–517, Atlanta, Georgia. Association for Computational Linguistics.

Chen, M., Ge, T., Zhang, X., Wei, F., and Zhou, M. (2020). Improving the efficiency of grammatical error correction with erroneous span detection and correction. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 7162–7169, Online. Association for Computational Linguistics.

Chodorow, M. and Leacock, C. (2000). An unsupervised method for detecting grammatical errors. In Proceedings of the Conference of the North American Chapter of the Association of Computational Linguistics (NAACL), pages 140––147. Association for Computational Linguistics.

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, pages 213–227, Florence, Italy. Association for Computational Linguistics.

Chollampatt, S. and Ng, H. T. (2018a). A multilayer convolutional encoder-decoder neural network for grammatical error correction. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence.

Chollampatt, S. and Ng, H. T. (2018b). A reassessment of reference-based grammatical error correction metrics. In Proceedings of the 27th International Conference on Computational Linguistics, pages 2730–2741, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Chollampatt, S., Wang, W., and Ng, H. T. (2019). Cross-sentence grammatical error correction. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 435–445, Florence, Italy. Association for Computational Linguistics.

Choshen, L. and Abend, O. (2018a). Automatic metric validation for grammatical error correction. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1372–1382, Melbourne, Australia. Association for Computational Linguistics.

Choshen, L. and Abend, O. (2018b). Reference-less measure of faithfulness for grammatical error correction. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), pages 124–129, New Orleans, Louisiana. Association for Computational Linguistics.

Choshen, L., Nikolaev, D., Berzak, Y., and Abend, O. (2020). Classifying syntactic errors in learner language. In Proceedings of the 24th Conference on Computational Natural Language Learning, pages 97–107, Online. Association for Computational Linguistics.

Dahlmeier, D. and Ng, H. T. (2011). Grammatical error correction with alternating structure optimization. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 915–923, Portland, Oregon, USA. Association for Computational Linguistics.

Dahlmeier, D. and Ng, H. T. (2012a). A beam-search decoder for grammatical error correction. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pages 568–578, Jeju Island, Korea. Association for Computational Linguistics.

Dahlmeier, D. and Ng, H. T. (2012b). Better evaluation for grammatical error correction. In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 568–572, Montréal, Canada. Association for Computational Linguistics.

Dahlmeier, D., Ng, H. T., and Wu, S. M. (2013). Building a large annotated corpus of learner English: The NUS corpus of learner english. In Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications, pages 22–31, Atlanta, Georgia. Association for Computational Linguistics.

Dale, R., Anisimoff, I., and Narroway, G. (2012). HOO 2012: A report on the preposition and determiner error correction shared task. In Proceedings of the Seventh Workshop on Building Educational Applications Using NLP, pages 54–62. Association for Computational Linguistics.

Dale, R. and Kilgarriff, A. (2011). Helping Our Own: The HOO 2011 pilot shared task. In Proceedings of the 13th European Workshop on Natural Language Generation, pages 242–249. Association for Computational Linguistics.

Daudaravicius, V., Banchs, R. E., Volodina, E., and Napoles, C. (2016). A report on the automatic evaluation of scientific writing shared task. In Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications, pages 53–62, San Diego, CA. Association for Computational Linguistics.

De Felice, R. and Pulman, S. G. (2008). A classifier-based approach to preposition and determiner error correction in l2 english. In Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pages 169–176, Manchester, UK. Coling 2008 Organizing Committee.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Domeij, R., Knutsson, O., Carlberger, J., and Kann, V. (2000). Granska - an efficient hybrid system for swedish grammar checking. In Proceedings of the 12th Nordic Conference in Computational Linguistics (Nodalida-99), pages 28––40. Department of Linguistics, University of Trondheim.

Fallman, D. (2002). The penguin: Using the web as a database for descriptive and dynamic grammar and spell checking. In CHI '02 Extended Abstracts on Human Factors in Computing Systems, CHI EA '02, pages 616–617, New York. ACM.

Felice, M. and Briscoe, T. (2015). Towards a standard evaluation method for grammatical error detection and correction. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 578–587, Denver, Colorado. Association for Computational Linguistics.

Felice, M., Yuan, Z., Andersen, Ø. E., Yannakoudakis, H., and Kochmar, E. (2014). Grammatical error correction using hybrid systems and type filtering. In Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task, pages 15–24, Baltimore, Maryland. Association for Computational Linguistics.

Gamon, M. (2010). Using mostly native data to correct errors in learners' writing. In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pages 163–171, Los Angeles, California. Association for Computational Linguistics.

Gamon, M. and Leacock, C. (2010). Search right and thou shalt find ... using web queries for learner error detection. In Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications, pages 37–44, Los Angeles, California. Association for Computational Linguistics.

Gamon, M., Leacock, C., Brockett, C., Dolan, W. B., Gao, J., Belenko, D., and Klementiev, A. (2009). Using statistical techniques and web search to correct esl errors. CALICO Journal, 26(3):491–511.

Ge, T., Wei, F., and Zhou, M. (2018a). Fluency boost learning and inference for neural grammatical error correction. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1055–1065, Melbourne, Australia. Association for Computational Linguistics.

Ge, T., Wei, F., and Zhou, M. (2018b). Reaching human-level performance in automatic grammatical error correction: An empirical study.

Gehring, J., Auli, M., Grangier, D., and Dauphin, Y. (2017). A convolutional encoder model for neural machine translation. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 123–135, Vancouver, Canada. Association for Computational Linguistics.

Grundkiewicz, R. and Junczys-Dowmunt, M. (2014). The wiked error corpus: A corpus of corrective wikipedia edits and its application to grammatical error correction. In Przepiórkowski, A. and Ogrodniczuk, M., editors, Advances in Natural Language Processing – Lecture Notes in Computer Science, volume 8686, pages 478–490. Springer.

Grundkiewicz, R. and Junczys-Dowmunt, M. (2018a). Near human-level performance in grammatical error correction with hybrid machine translation. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), pages 284–290. Association for Computational Linguistics.

Grundkiewicz, R. and Junczys-Dowmunt, M. (2018b). Near human-level performance in grammatical error correction with hybrid machine translation. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), pages 284–290, New Orleans, Louisiana. Association for Computational Linguistics.

Grundkiewicz, R. and Junczys-Dowmunt, M. (2019). Minimally-augmented grammatical error correction. In Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019), pages 357–363, Hong Kong, China. Association for Computational Linguistics.

Grundkiewicz, R., Junczys-Dowmunt, M., and Gillian, E. (2015). Human evaluation of grammatical error correction systems. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 461–470, Lisbon, Portugal. Association for Computational Linguistics.

Grundkiewicz, R., Junczys-Dowmunt, M., and Heafield, K. (2019). Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, pages 252–263, Florence, Italy. Association for Computational Linguistics.

Heidorn, G., Jensen, K., Miller, L., Byrd, R., and Chodorow, M. (1982). The epistle text-critiquing system. IBM Systems Journal, 21(3):305–326.

Heilman, M., Cahill, A., Madnani, N., Lopez, M., Mulholland, M., and Tetreault, J. (2014). Predicting grammaticality on an ordinal scale. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 174–180, Baltimore, Maryland. Association for Computational Linguistics.

Hermet, M. and Désilets, A. (2009). Using first and second language models to correct preposition errors in second language authoring. In Proceedings of the Fourth Workshop on Innovative Use of NLP for Building Educational Applications, EdAppsNLP '09, pages 64–72, Boulder, Colorado. Association for Computational Linguistics.

Hermet, M., Désilets, A., and Szpakowicz, S. (2008). Using the web as a linguistic resource to automatically correct lexico-syntactic errors. In Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., and Tapias, D., editors, Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), pages 874–878, Marrakech, Morocco. European Language Resources Association (ELRA).

Islam, A. and Inkpen, D. (2011). Correcting different types of errors in texts. In Proceedings of the 24th Canadian Conference on Advances in Artificial Intelligence, Canadian AI'11, pages 192–203, Berlin, Heidelberg. Springer-Verlag.

Ji, J., Wang, Q., Toutanova, K., Gong, Y., Truong, S., and Gao, J. (2017). A nested attention neural hybrid model for grammatical error correction. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 753–762. Association for Computational Linguistics.

Jia, Z., Wang, P., and Zhao, H. (2013). Grammatical error correction as multiclass classification with single model. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task, pages 74–81, Sofia, Bulgaria. Association for Computational Linguistics.

Junczys-Dowmunt, M. and Grundkiewicz, R. (2014). The amu system in the conll-2014 shared task: Grammatical error correction by data-intensive and feature-rich statistical machine translation. In Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task, pages 25–33, Baltimore, Maryland. Association for Computational Linguistics.

Junczys-Dowmunt, M. and Grundkiewicz, R. (2016). Log-linear combinations of monolingual and bilingual neural machine translation models for automatic post-editing. In Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers, pages 751–758, Berlin, Germany. Association for Computational Linguistics.

Junczys-Dowmunt, M. and Grundkiewicz, R. (2018). MS-UEdin submission to the WMT2018 APE shared task: Dual-source transformer for automatic post-editing. In Proceedings of the Third Conference on Machine Translation: Shared Task Papers, pages 822–826, Belgium, Brussels. Association for Computational Linguistics.

Junczys-Dowmunt, M., Grundkiewicz, R., Guha, S., and Heafield, K. (2018). Approaching neural grammatical error correction as a low-resource machine translation task. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 595–606, New Orleans, Louisiana. Association for Computational Linguistics.

Junczys-Dowmunt, M. and Junczys-Dowmunt, M. (2017). The AMU-UEdin submission to the WMT 2017 shared task on automatic post-editing. In Proceedings of the Second Conference on Machine Translation, pages 639–646, Copenhagen, Denmark. Association for Computational Linguistics.

Kaneko, M., Mita, M., Kiyono, S., Suzuki, J., and Inui, K. (2020). Encoder-decoder models can benefit from pre-trained masked language models in grammatical error correction. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4248–4254, Online. Association for Computational Linguistics.

Kasewa, S., Stenetorp, P., and Riedel, S. (2018). Wronging a right: Generating better errors to improve grammatical error detection. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 4977–4983, Brussels, Belgium. Association for Computational Linguistics.

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

Koehn, P. (2020). Neural Machine Translation. Cambridge University Press.

Leacock, C. and Chodorow, M. (2003). Automated grammatical error detection, pages 195––207. Lawrence Erlbaum Associates.

Lee, J. (2004). Automatic article restoration. In Susan Dumais, D. M. and Roukos, S., editors, HLT-NAACL 2004: Student Research Workshop, pages 31–36, Boston, Massachusetts, USA. Association for Computational Linguistics.

Li, Y., Anastasopoulos, A., and Black, A. W. (2020). Towards minimal supervision bert-based grammar error correction (student abstract). Proceedings of the AAAI Conference on Artificial Intelligence, 34(10):13859–13860.

Lichtarge, J., Alberti, C., Kumar, S., Shazeer, N., Parmar, N., and Tong, S. (2019). Corpora generation for grammatical error correction. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 3291–3301, Minneapolis, Minnesota. Association for Computational Linguistics.

Lin, C.-J. and Chen, S.-H. (2015). Ntou chinese grammar checker for cged shared task. In Proceedings of the 2nd Workshop on Natural Language Processing Techniques for Educational Applications, pages 15–19, Beijing, China. Association for Computational Linguistics.

Macdonald, N., Frase, L., Gingrich, P., and Keenan, S. (1982). The writer's workbench: Computer aids for text analysis. Communications, IEEE Transactions on, 30(1):105–110.

Madnani, N., Tetreault, J., and Chodorow, M. (2012). Exploring grammatical error correction with not-so-crummy machine translation. In Proceedings of the Seventh Workshop on Building Educational Applications Using NLP, pages 44–53, Montreal, Canada. Association for Computational Linguistics.

Mangu, L. and Brill, E. (1997). Automatic rule acquisition for spelling correction. In Proceedings of the Fourteenth International Conference on Machine Learning, ICML '97, pages 187–194, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Miceli Barone, A. V., Helcl, J., Sennrich, R., Haddow, B., and Birch, A. (2017). Deep architectures for neural machine translation. In Proceedings of the Second Conference on Machine Translation, pages 99–107, Copenhagen, Denmark. Association for Computational Linguistics.

Mizumoto, T., Komachi, M., Nagata, M., and Matsumoto, Y. (2011). Mining revision log of language learning SNS for automated Japanese error correction of second language learners. In Proceedings of 5th International Joint Conference on Natural Language Processing, pages 147–155. Asian Federation of Natural Language Processing.

Mohit, B., Rozovskaya, A., Habash, N., Zaghouani, W., and Obeid, O. (2014). The first qalb shared task on automatic text correction for arabic. In Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP), pages 39–47, Doha, Qatar. Association for Computational Linguistics.

Náplava, J. and Straka, M. (2019). Grammatical error correction in low-resource scenarios. In Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019), pages 346–356, Hong Kong, China. Association for Computational Linguistics.

Napoles, C., Nădejde, M., and Tetreault, J. (2019). Enabling robust grammatical error correction in new domains: Data sets, metrics, and analyses. Transactions of the Association for Computational Linguistics, 7:551–566.

Napoles, C., Sakaguchi, K., Post, M., and Tetreault, J. (2015). Ground truth for grammatical error correction metrics. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 588–593, Beijing, China. Association for Computational Linguistics.

Napoles, C., Sakaguchi, K., Post, M., and Tetreault, J. R. (2016). GLEU without tuning. CoRR, abs/1605.02592.

Napoles, C., Sakaguchi, K., and Tetreault, J. (2017). JFLEG: a fluency corpus and benchmark for grammatical error correction. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, pages 229–234, Valencia, Spain. Association for Computational Linguistics.

Ng, H. T., Wu, S. M., Briscoe, T., Hadiwinoto, C., Susanto, R. H., and Bryant, C. (2014). The CoNLL-2014 shared task on grammatical error correction. In Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task, pages 1–14. Association for Computational Linguistics.

Ng, H. T., Wu, S. M., Wu, Y., Hadiwinoto, C., and Tetreault, J. (2013). The CoNLL-2013 shared task on grammatical error correction. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task, pages 1–12. Association for Computational Linguistics.

Nicholls, D. (2003). The cambridge learner corpus: Error coding and analysis for lexicography and ELT. In Proceedings of the Corpus Linguistics 2003 conference, pages 572–581.

Okanohara, D. and Tsujii, J. (2007). A discriminative language model with pseudo-negative samples. In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pages 73–80, Prague, Czech Republic. Association for Computational Linguistics.

Omelianchuk, K., Atrasevych, V., Chernodub, A., and Skurzhanskyi, O. (2020). GECToR – grammatical error correction: Tag, not rewrite. In Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications, pages 163–170, Seattle, WA, USA â†’ Online. Association for Computational Linguistics.

Park, Y. A. and Levy, R. (2011). Automated whole sentence grammar correction using a noisy channel model. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 934–944, Portland, Oregon, USA. Association for Computational Linguistics.

Quan, L., Kolomiyets, O., and Moens, M.-F. (2012). Ku leuven at hoo-2012: A hybrid approach to detection and correction of determiner and preposition errors in non-native english text. In Proceedings of the Seventh Workshop on Building Educational Applications Using NLP, pages 263–271, Montréal, Canada. Association for Computational Linguistics.

Rei, M., Felice, M., Yuan, Z., and Briscoe, T. (2017). Artificial error generation with machine translation and syntactic patterns. In Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications, pages 287–292, Copenhagen, Denmark. Association for Computational Linguistics.

Rei, M. and Yannakoudakis, H. (2016). Compositional sequence labeling models for error detection in learner writing. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1181–1191, Berlin, Germany. Association for Computational Linguistics.

Richardson, S. D. and Braden-Harder, L. C. (1988). The experience of developing a large-scale natural language text processing system: Critique. In Proceedings of the Second Conference on Applied Natural Language Processing, ANLC '88, pages 195–202, Stroudsburg, PA, USA. Association for Computational Linguistics.

Rozovskaya, A., Bouamor, H., Habash, N., Zaghouani, W., Obeid, O., and Mohit, B. (2015). The second qalb shared task on automatic text correction for arabic. In Proceedings of the Second Workshop on Arabic Natural Language Processing, pages 26–35, Beijing, China. Association for Computational Linguistics.

Rozovskaya, A., Chang, K.-W., Sammons, M., and Roth, D. (2013). The university of illinois system in the conll-2013 shared task. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task, pages 13–19, Sofia, Bulgaria. Association for Computational Linguistics.

Rozovskaya, A., Chang, K.-W., Sammons, M., Roth, D., and Habash, N. (2014). The illinois-columbia system in the conll-2014 shared task. In Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task, pages 34–42, Baltimore, Maryland. Association for Computational Linguistics.

Rozovskaya, A. and Roth, D. (2011). Algorithm selection and model adaptation for esl correction tasks. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 924–933. Association for Computational Linguistics.

Rozovskaya, A. and Roth, D. (2019). Grammar error correction in morphologically-rich languages: The case of Russian. Transactions of the Association for Computational Linguistics, 7:1–17.

Rozovskaya, A., Sammons, M., Gioja, J., and Roth, D. (2011). University of illinois system in hoo text correction shared task. In Proceedings of the Generation Challenges Session at the 13th European Workshop on Natural Language Generation, pages 263–266, Nancy, France. Association for Computational Linguistics.

Sakaguchi, K., Hayashibe, Y., Kondo, S., Kanashiro, L., Mizumoto, T., Komachi, M., and Matsumoto, Y. (2012). Naist at the hoo 2012 shared task. In Proceedings of the Seventh Workshop on Building Educational Applications Using NLP, pages 281–288, Montréal, Canada. Association for Computational Linguistics.

Sakaguchi, K., Post, M., and Van Durme, B. (2017). Grammatical error correction with neural reinforcement learning. In Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 366–372, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Shannon, C. E. (1948). A Mathematical Theory of Communication. The Bell System Technical Journal, 27(3):379–423.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15(56):1929–1958.

Stahlberg, F. (2020). Neural machine translation: A review. Journal of Artificial Intelligence Research, 69:343–418.

Stahlberg, F., Bryant, C., and Byrne, B. (2019). Neural Grammatical Error Correction with Finite State Transducers. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4033–4039, Minneapolis, Minnesota. Association for Computational Linguistics.

Stahlberg, F. and Kumar, S. (2020). Seq2Edits: Sequence transduction using span-level edit operations. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 5147–5159, Online. Association for Computational Linguistics.

Sun, C., Jin, X., Lin, L., Zhao, Y., and Wang, X. (2015). Convolutional neural networks for correcting english article errors. In Li, J., Ji, H., Zhao, D., and Feng, Y., editors, Natural Language Processing and Chinese Computing, pages 102–110, Cham. Springer International Publishing.

Sun, Y. and Jiang, H. (2019). Contextual text denoising with masked language model. In Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019), pages 286–290, Hong Kong, China. Association for Computational Linguistics.

Tajiri, T., Komachi, M., and Matsumoto, Y. (2012). Tense and aspect error correction for ESL learners using global context. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 198–202. Association for Computational Linguistics.

Tetreault, J. and Chodorow, M. (2009). Examining the use of region web counts for esl error detection. In Proceedings of the Web as Corpus Workshop (WAC-5). Elhuyar Fundazioa.

van den Bosch, A. and Berck, P. (2013). Memory-based grammatical error correction. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task, pages 102–108, Sofia, Bulgaria. Association for Computational Linguistics.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, Advances in Neural Information Processing Systems, volume 30, pages 5998–6008. Curran Associates, Inc.

Wagner, J., Foster, J., and van Genabith, J. (2007). A comparative evaluation of deep and shallow approaches to the automatic detection of common grammatical errors. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pages 112–121, Prague, Czech Republic. Association for Computational Linguistics.

Wang, P., Jia, Z., and Zhao, H. (2014). Grammatical error detection and correction using a single maximum entropy model. In Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task, pages 74–82, Baltimore, Maryland. Association for Computational Linguistics.

West, R., Park, Y. A., and Levy, R. (2011). Bilingual random walk models for automated grammar correction of esl author-produced text. In Proceedings of the Sixth Workshop on Innovative Use of NLP for Building Educational Applications, pages 170–179, Portland, Oregon. Association for Computational Linguistics.

White, M. and Rozovskaya, A. (2020). A comparative study of synthetic data generation methods for grammatical error correction. In Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications, pages 198–208, Seattle, WA, USA â†’ Online. Association for Computational Linguistics.

Wilcox-O'Hearn, L. A. (2013). A noisy channel model framework for grammatical correction. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task, pages 109–114, Sofia, Bulgaria. Association for Computational Linguistics.

Wu, J.-C., Chang, Y.-C., Mitamura, T., and Chang, J. S. (2010). Automatic collocation suggestion in academic writing. In Proceedings of the ACL 2010 Conference Short Papers, pages 115–119, Uppsala, Sweden. Association for Computational Linguistics.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, L., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., and Dean, J. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. CoRR, abs/1609.08144.

Xie, W., Huang, P., Zhang, X., Hong, K., Huang, Q., Chen, B., and Huang, L. (2015). Chinese spelling check system based on n-gram model. In Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing, pages 128–136, Beijing, China. Association for Computational Linguistics.

Xie, Z., Avati, A., Arivazhagan, N., Jurafsky, D., and Ng, A. Y. (2016). Neural language correction with character-based attention.

Xie, Z., Genthial, G., Xie, S., Ng, A., and Jurafsky, D. (2018). Noising and denoising natural language: Diverse backtranslation for grammar correction. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 619–628, New Orleans, Louisiana. Association for Computational Linguistics.

Yannakoudakis, H., Briscoe, T., and Medlock, B. (2011). A new dataset and method for automatically grading ESOL texts. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 180–189, Portland, Oregon, USA. Association for Computational Linguistics.

Yannakoudakis, H., Rei, M., Andersen, Ø. E., and Yuan, Z. (2017). Neural sequence-labelling models for grammatical error correction. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 2795–2806, Copenhagen, Denmark. Association for Computational Linguistics.

Yannakoudakis, H., Øistein E Andersen, Geranpayeh, A., Briscoe, T., and Nicholls, D. (2018). Developing an automated writing placement system for esl learners. Applied Measurement in Education, 31(3):251–267.

Yin, F., Long, Q., Meng, T., and Chang, K.-W. (2020). On the robustness of language encoders against grammatical errors. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 3386–3403, Online. Association for Computational Linguistics.

Yoshimoto, I., Kose, T., Mitsuzawa, K., Sakaguchi, K., Mizumoto, T., Hayashibe, Y., Komachi, M., and Matsumoto, Y. (2013). Naist at 2013 conll grammatical error correction shared task. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task, pages 26–33, Sofia, Bulgaria. Association for Computational Linguistics.

Yuan, Z. and Briscoe, T. (2016). Grammatical error correction using neural machine translation. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 380–386. Association for Computational Linguistics.

Yuan, Z., Briscoe, T., and Felice, M. (2016). Candidate re-ranking for smt-based grammatical error correction. In Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications, pages 256–266, San Diego, CA. Association for Computational Linguistics.

Zhang, L. and Wang, H. (2014). A unified framework for grammar error correction. In Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task, pages 96–102, Baltimore, Maryland. Association for Computational Linguistics.

Zhang, S., Huang, H., Liu, J., and Li, H. (2020). Spelling error correction with soft-masked BERT. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 882–890, Online. Association for Computational Linguistics.

Zhao, W., Wang, L., Shen, K., Jia, R., and Liu, J. (2019). Improving grammatical error correction via pre-training a copy-augmented architecture with unlabeled data. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 156–165, Minneapolis, Minnesota. Association for Computational Linguistics.

Zhao, Y., Jiang, N., Sun, W., and Wan, X. (2018). Overview of the NLPCC 2018 shared task: Grammatical error correction. In Zhang, M., Ng, V., Zhao, D., Li, S., and Zan, H., editors, Natural Language Processing and Chinese Computing - 7th CCF International Conference, NLPCC 2018, Hohhot, China, August 26-30, 2018, Proceedings, Part II, volume 11109 of Lecture Notes in Computer Science, pages 439–445. Springer.

Zhou, W., Ge, T., Mu, C., Xu, K., Wei, F., and Zhou, M. (2020). Improving grammatical error correction with machine translation pairs. In Findings of the Association for Computational Linguistics: EMNLP 2020, pages 318–328, Online. Association for Computational Linguistics.