A Web-based English Proofing System for English as a Second Language Users

Xing Yi[1], **Jianfeng Gao**[2] **and William B. Dolan**[2]
[1]Center for Intelligent Information Retrieval, Department of Computer Science
University of Massachusetts, Amherst, MA 01003-4610, USA
yixing@cs.umass.edu
[2]Microsoft Research, One Microsoft Way, Redmond, WA 98052, USA
{jfgao,billdol}@microsoft.com

Abstract

We describe an algorithm that relies on web frequency counts to identify and correct writing errors made by non-native writers of English. Evaluation of the system on a real-world ESL corpus showed very promising performance on the very difficult problem of critiquing English determiner use: 62% precision and 41% recall, with a false flag rate of only 2% (compared to a random-guessing baseline of 5% precision, 7% recall, and more than 80% false flag rate). Performance on collocation errors was less good, suggesting that a web-based approach should be combined with local linguistic resources to achieve both effectiveness and efficiency.

1 Introduction

Proofing technology for native speakers of English has been a focus of work for decades, and some tools like spell checkers and grammar checkers have become standard features of document processing software products. However, designing an English proofing system for English as a Second Language (ESL) users presents a major challenge: ESL writing errors vary greatly among users with different language backgrounds and proficiency levels. Recent work by Brockett *et al.* (2006) utilized phrasal Statistical Machine Translation (SMT) techniques to correct ESL writing errors and demonstrated that this data-intensive SMT approach is very promising, but they also pointed out SMT approach relies on the availability of large amount of training data. The expense and difficulty of collecting large quantities of raw and edited ESL prose pose an obstacle to this approach.

Search Phrase	Google.com	Live.com	Yahoo.com
English as Second Language	306,000	52,407	386,000
English as ***a*** Second Language	1,490,000	38,336,308	4,250,000

Table 1: Web Hits for Phrasal Usages

In this work we consider the prospect of using the Web, with its billions of web pages, as a data source with the potential to aid ESL writers. Our research is motivated by the observation that ESL users already use the Web as a corpus of good English, often using search engines to decide whether a particular spelling, phrase, or syntactic construction is consistent with usage found on the Web. For example, unsure whether the native-sounding phrase includes the determiner "a", a user might search for both quoted strings "English as Second Language" and "English as a Second Language". The counts obtained for each of these phrases on three different search engines are shown in Table 1. Note the correct version, "English as a Second Language", has a much higher number of web hits.

In order to determine whether this approach holds promise, we implemented a web-based system for ESL writing error proofing. This pilot study was intended to:

1. identify different types of ESL writing errors and how often they occur in ESL users' writing samples, so that the challenges and difficulties of ESL error proofing can be understood better;

2. explore the advantages and drawbacks of a web-

based approach, discover useful web data features, and identify which types of ESL errors can be reliably proofed using this technique.

We first catalog some major categories of ESL writing errors, then review related work. Section 3 describes our Web-based English Proofing System for ESL users (called **ESL-WEPS** later). Section 4 presents experimental results. Section 5 concludes.

1.1 ESL Writing Errors

In order to get ESL writing samples, we employed a third party to identify large volumes of ESL web pages (mostly from Japanese, Korean and Chinese ESL users' blogs), and cull 1K non-native sentences. A native speaker then rewrote these ESL sentences – when possible – to produce a native-sounding version. 353 (34.9%) of the original 1012 ESL sentences were labeled "native-like", another 347 (34.3%) were rewritten, and the remaining 312 (30.8%) were classified as simply unintelligible.

Table 2 shows some examples from the corpus illustrating some typical types of ESL writing errors involving: (1) Verb-Noun Collocations (VNC) and (4) Adjective-Noun Collocations (ANC); (2) incorrect use of the transitive verb "attend"; (3) determiner (article) usage problems; and (5) more complex lexical and style problems. We analyzed all the pre- and post-edited ESL samples and found 441 ESL errors: about 20% are determiner usage problems(missing/extra/misused); 15% are VNC errors, 1% are ANC errors; others represent complex syntactic, lexical or style problems. Multiple errors can co-occur in one sentence. These show that real-world ESL error proofing is very challenging.

Our findings are consistent with previous research results on ESL writing errors in two respects:

1. ESL users have significantly more problems with determiner usage than native speakers because the use and omission of definite and indefinite articles varies across different languages (Schneider and McCoy, 1998)(Lonsdale and Strong-Krause, 2003).

2. Collocation errors are common among ESL users, and collocational knowledge contributes to the difference between native speakers and ESL learners (Shei and Pain, 2000): in CLEC, a real-world Chinese English Learner Corpus (Gui and Yang, 2003), about 30% of ESL writing errors involve different types of collocation errors.

In the remainder of the paper, we focus on proofing determiner usage and VNC errors.

2 Related Work

Researchers have recently proposed some successful learning-based approaches for the determiner selection task (Minnen et al., 2000), but most of this work has aimed only at helping native English users correct typographical errors. Gamon *et al.*(2008) recently addressed the challenging task of proofing writing errors for ESL users: they propose combining contextual speller techniques and language modeling for proofing several types of ESL errors, and demonstrate some promising results. In a departure from this work, our system directly uses web data for the ESL error proofing task.

There is a small body of previous work on the use of online systems aimed at helping ESL learners correct collocation errors. In Shei and Pain's system (2000), for instance, the *British National Corpus (BNC)* is used to extract English collocations, and an ESL learner writing corpus is then used to build a collocation Error Library. In Jian *et al.*'s system (2004), the *BNC* is also used as a source of collocations, with collocation instances and translation counterparts from the bilingual corpus identified and shown to ESL users. In contrast to this earlier work, our system uses the web as a corpus, with string frequency counts from a search engine index used to indicate whether a particular collocation is being used correctly.

3 Web-based English Proofing System for ESL Users (ESL-WEPS)

The architecture of **ESL-WEPS**, which consists of four main components, is shown in Fig.1.

Parse ESL Sentence and Identify Check Points ESL-WEPS first tags and chunks (Sang and Buckholz, 2000) the input ESL sentence[1], and identifies the elements of the structures in the sentence to be checked according to certain heuristics: when

[1]One in-house HMM chunker trained on English Penn Treebank was used.

ID	Pre-editing version	Post-editing version
1	Which team can ***take*** the champion?	Which team will ***win*** the championship?
2	I ***attend to*** Pyoung Taek University.	I ***attend*** Pyoung Taek University.
3	***I'm a Japanese*** and studying Info and Computer Science at Keio University.	***I'm Japanese*** and studying Info Computer Science at Keio University.
4	Her works are ***kinda*** erotic but they will never arouse any obscene, ***devil thoughts*** which might destroy the soul of the designer.	Her works are ***kind of*** erotic, but they will never arouse any obscene, ***evil thoughts*** which might destroy the soul of the designer.
5	I think it is so beautiful to ***go the way of theology and very attractive too, especially in the area of Christianity***.	I think it is so beautiful to ***get into theology, especially Christianity, which attracts me***.

Table 2: Some pre- and post-editing ESL writing samples, Bold Italic characters show where the ESL errors are and how they are corrected/rewritten by native English speaker.

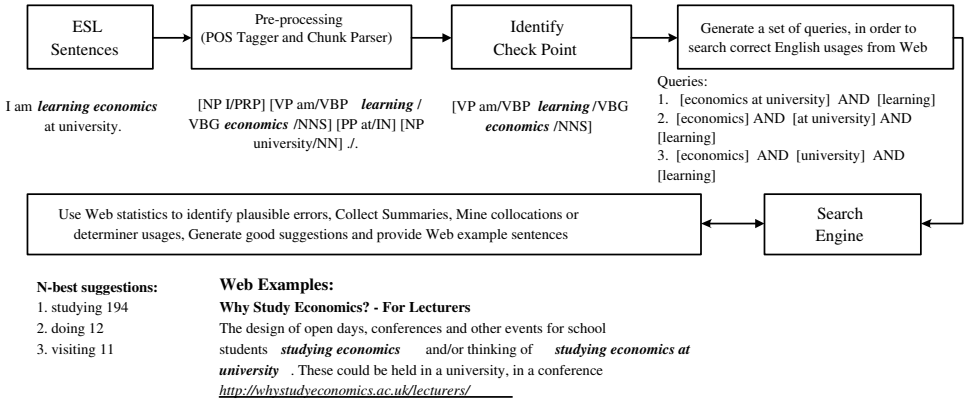

Figure 1: System Architecture

checking VNC errors, the system searches for a structure of the form (VP)(NP) or (VP)(PP)(NP) in the chunked sentence; when checking determiner usage, the system searches for (NP). Table 3 shows some examples. For efficiency and effectiveness, the user can specify that only one specific error type be critiqued; otherwise it will check both error types: first determiner usage, then collocations.

Generate Queries In order to find appropriate web examples, ESL-WEPS generates at each check point a set of queries. These queries involve three different granularity levels, according to sentence's syntax structure:

1. **Reduced Sentence Level**. In order to use more contextual information, our system preferentially generates a maximal-length query hereafter called *S-Queries*, by using the original sentence. For the check point chunk, the verb/adj. to be checked is found and extracted based on POS tags; other chunks are simply concatenated and used to formulate the query. For example, for the first example in Table 3, the S-Query is ['I have' AND 'this person for years' AND 'recognized'].

2. **Chunk Level**. The system segments each ESL sentence according to chunk tags and utilizes chunk pairs to generate a query, hereafter referred to as a *C-Query*, e.g. the C-Query for the second example in Table 3 is ['I' AND 'went' AND 'to climb' AND 'a tall mountain' AND 'last week']

3. **Word Level**. The system generates queries by using keywords from the original string, in the processing eliminating stopwords used in typical IR engines, hereafter referred to as a *W-Query*, e.g. W-Query for the first example in Table 3 is ['I' AND 'have' AND 'person' AND 'years' AND 'recognized']

As queries get longer, web search engines tend to return fewer and fewer results. Therefore, ESL-WEPS first segments the original ESL sentence by using punctuation characters like commas and semicolons, then generates a query from only the part which contains the given check point. When checking determiner usage, three different cases (a or an/the/none)

Parsed ESL sentence	Error Type	Check Points
(NP I/PRP) (VP have/VBP recognized/VBN) (NP this/DT person/NN) (PP for/IN) (NP years/NNS) ./.	VNC	recognized this person
(NP I/PRP) (VP went/VBD) (VP to/TO climb/VB) (NP a/DT tall/JJ mountain/NN) (NP last/JJ week/NN) ./.	ANC	tall mountain, last week
(NP I/PRP) (VP went/VBD) (PP to/TO) (NP coffee/NN) (NP shop/NN) (NP yesterday/NN) ./.	Determiner usage	coffee, shop, yesterday
(NP Someone/NN) (ADVP once/RB) (VP said/VBD) (SBAR that/IN) (ADVP when/WRB) (NP you/PRP) (VP meet/VBP) (NP a/DT right/JJ person/NN) (PP at/IN) (NP the/DT wrong/JJ time/NN),/, (NP it/PRP) (VP 's/VBZ) (NP a/DT pity/NN)./.	Determiner usage	meet a right person at the wrong time 's a pity

Table 3: Parsed ESL sentences and Check Points.

are considered for each check point. For instance, given the last example in Table 3, three C-Queries will be generated: [meet a right person],[meet the right person] and [meet right person]. Note that a term which has been POS-tagged as NNP (proper noun) will be skipped and not used for generating queries in order to obtain more web hits.

Retreive Web Statistics, Collect Snippets To collect enough web examples, three levels of query sets are submitted to the search engine in the following order: S-Query, C-Query, and finally W-Query. For each query, the web hits *df* returned by search engine is recorded, and the snippets from the top 1000 hits are collected. For efficiency reasons, we follow Dumais (2002)'s approach: the system relies only on snippets rather than full-text of pages returned for each hit; and does not rely on parsing or POS-tagging for this step. However, a lexicon is used in order to determine the possible parts-of-speech of a word as well as its morphological variants. For example, to find the correct VNC for a given noun 'tea' in the returned snippets, the verb **drank** in the same clause will be matched before 'tea'.

Identify Errors and Mine Correct Usages To detect determiner usage errors, both the web hit df_q and the length l_q of a given query q are utilized, since longer query phrases usually lead to fewer web hits. DFL_q, $DFLMAX$, q_{max} and R_q are defined as:

$$DFL_q = df_q \times l_q, for\ a\ given\ query\ q;$$
$$DFLMAX = \max(DFL_q),$$
$$q_{max} = \arg\max_q(DFL_q),$$
$$q \in \{queries\ for\ a\ given\ check\ point\};$$
$$R_q = DFL_q/DFLMAX, given\ query\ q\ and\ check\ point.$$

If $DFLMAX$ is less than a given threshold t_1, this check point will be skipped; otherwise the q_{max} indicates the best usage. We also calculate the relative ratio R_q for three usages (a or an/the/none). If R_q is larger than a threshold t_2 for a query q, the system will not report that usage as an error because it is sufficiently supported by web data.

For collocation check points, ESL-WEPS may interact twice with the search engine: first, it issues query sets to collect web examples and identify plausible collocation errors; then, if errors are detected, new query sets will be issued in the second step in order to mine correct collocations from new web examples. For example, for the first sentence in Table 3, the S-Query will be ['I have' AND 'this person for years' AND 'recognized']; the system analyzes returned snippets and identifies 'recognized' as a possible error. The system then issues a new S-Query ['I have' AND 'this person for years'], and finally mines the new set of snippets to discover that 'known' is the preferred lexical option. In contrast to proofing determiner usages errors, $mfreq$:

$$mfreq = frequency\ of\ matched\ collocational\ verb/adj.\ in\ the\ snippets\ for\ a\ given\ noun,$$

is utilized to both identify errors and suggest correct VNCs/ANCs. If $mfreq$ is larger than a threshold t_3, the system will conclude that the collocation is plausible and skip the suggestion step.

4 Experiments

In order to evaluate the proofing algorithm described above, we utilized the MSN search engine API and the ESL writing sample set described in Section 1.1 to evaluate the algorithm's performance on two tasks: determiner usage and VNC proofing. From a practical standpoint, we consider precision on the proofing task to be considerably more important than recall: false flags are annoying and highly visible to the user, while recall failures are much less problematic.

Given the complicated nature of the ESL error proofing task, about 60% of ESL sentences in our set that contained determiner errors also contained other types of ESL errors. As a result, we were forced to slightly revise the typical precision/recall measurement in order to evaluate performance. First,

Good Proofing Examples	
Error sentence 1	In my opinion, therefore, when we describe terrorism, its crucially important that we ***consider the degree of the influence*** (i.e., power) on the other countries.
proofing suggestion	consider the degree of influence
Error sentence 2	Someone once said that when you ***meet a right person at the wrong time***, it's a pity.
proofing suggestion	meet the right person at the wrong time
Plausible Useful Proofing Examples	
Error sentence 3	The most powerful place in Beijing, and ***in the whole China***.
native speaker suggestion	in the whole of China
system suggestion	in whole China
Error sentence 4	Me, I wanna keep in touch with old friends and wanna talk with anyone who ***has different thought***, etc.
native speaker suggestion	has different ideas
system suggestion	has a different thought

Table 4: ESL Determiner Usage Proofing by Native Speaker and ESL-WEPS.

Good Proofing Examples	
Error sentence 1	I had great time there and ***got many friends***.
proofing suggestion	made many friends
Error sentence 2	Which team can ***take the champion***?
proofing suggestion	win the champion
Plausible Useful Proofing Examples	
Error sentence 3	It may ***sounds fun*** if I say my firm resolution of this year is to get a girl friend.
native speaker suggestion	sound funny
system suggestion	make * fun or get * fun

Table 5: ESL VNC Proofing by Native Speaker and ESL-WEPS.

we considered three cases: (1) the system correctly identifies an error and proposes a suggestion that exactly matches the native speaker's rewrite; (2) the system correctly identifies an error but makes a suggestion that differs from the native speaker's edit; and (3) the system incorrectly identifies an error. In the first case, we consider the proofing *good*, in the second, *plausibly useful*, and in the third case it is simply *wrong*. Correspondingly, we introduce the categories *Good Precision (GP)*, *Plausibly Useful Precision (PUP)* and *Error Suggestion Rate (ESR)*, which were calculated by:

$$GP = \frac{\#\ of\ Good\ Proofings}{\#\ of\ System's\ Proofings};$$
$$PUP = \frac{\#\ of\ Plausibly\ Useful\ Proofings}{\#\ of\ System's\ Proofings};$$
$$ESR = \frac{\#\ of\ Wrong\ Proofings}{\#\ of\ System's\ Proofings};$$
$$GP + PUP + ESR = 1$$

Furthermore, assuming that there are overall N_A errors for a given type A of ESL error , the typical *recall* and *false alarm* were calculated by:

$$recall = \frac{\#\ of\ Good\ Proofings}{N_A};$$
$$false\ alarm = \frac{\#\ of\ Wrong\ Proofings}{\#\ of\ Check\ points\ for\ ESL\ error\ A}$$

Table 4 and Table 5 show examples of ***Good*** or ***Plausibly Useful*** proofing for determiner usage and collocation errors, respectively. It can be seen the system makes plausibly useful proofing suggestions because some errors types are out of current system's checking range.

The system achieved very promising performance despite the fact that many of the test sentences contained other, complex ESL errors: using appropriate system parameters, ESL-WEPS showed recall 40.7% on determiner usage errors, with 62.5% of these proofing suggestions exactly matching the rewrites provided by native speakers. Crucially, the false flag rate was only 2%. Note that a random-guessing baseline was about 5% precision, 7% recall, but more than 80% false flag rate.

For collocation errors, we focused on the most common VNC proofing task. $mfreq$ and threshold t_3 described in Section 3 are used to control false alarm, GP and recall. A smaller t_3 can reduce recall, but can increase GP. Table 7 shows how performance changes with different settings for t_3, and Fig. 2(b) plots the GP/recall curve. Results are not very good: as recall increases, GP decreases too quickly, so that at 30.7% recall, precision is only 37.3%. We attribute this to the fact that most search engines only return the top 1000 web snippets for each query and our current system relies on this limited number of snippets to generate and rank candidates.

Recall	16.3%	30.2%	**40.7%**	44.2%	47.7%	50.0%
GP	73.7%	70.3%	**62.5%**	56.7%	53.3%	52.4%
PUP	15.8%	16.2%	**25.0%**	29.9%	29.9%	29.3%
false alarm	0.4%	1.4%	**2.0%**	2.6%	3.7%	4.3%

Table 6: Proofing performance of determiner usage changes when setting different system parameters.

Recall	11.3%	12.9%	17.8%	25.8%	29.0%	30.7%
GP	77.8%	53.3%	52.4%	43.2%	40.9%	37.3%
PUP	11.11%	33.33%	33.33%	45.10%	48.65%	50.00%
false alarm	0.28%	0.57%	0.85%	0.85%	1.13%	2.55%

Table 7: VNC Proofing performance changes when setting different system parameters.

5 Conclusion

This paper introduced an approach to the challenging real-world ESL writing error proofing task that uses the index of a web search engine for corpus statistics. We validated ESL-WEPS on a web-crawled ESL writing corpus and compared the system's proofing suggestions to those produced by native English speakers. Promising performance was achieved for proofing determiner errors, but less good results for VNC proofing, possibly because the current system uses web snippets to rank and generate collocation candidates. We are currently investigating a modified strategy that exploits high quality local collocation/synonym lists to limit the number of proposed Verb/Adj. candidates.

We are also collecting more ESL data to validate our system and are extending our system to more ESL error types. Recent experiments on new data showed that ESL-WEPS can also effectively proof incorrect choices of prepositions. Later research will compare the web-based approach to conventional corpus-based approaches like Gamon *et al.* (2008), and explore their combination to address complex ESL errors.

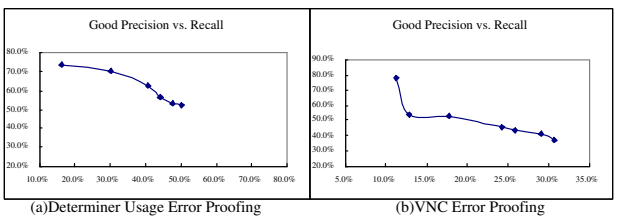

Figure 2: GP/recall curves. X and Y axis denotes GP and Recall respectively.

Acknowledgement The authors have benefited extensively from discussions with Michael Gamon and Chris Brockett. We also thank the Butler Hill Group for collecting the ESL examples.

References

C. Brockett, W. B. Dolan, and M. Gamon. 2006. Correcting ESL errors using phrasal smt techniques. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL*, pages 249–256, Sydney, Australia.

S. Dumais, M. Banko, E. Brill, J. Lin, and A. Ng. 2002. Web question answering: is more always better? In *Proceedings of the 25th Annual International ACM SIGIR*, pages 291–298, Tampere, Finland.

M. Gamon, J.F. Gao, C. Brockett, A. Klementiev, W.B. Dolan, and L. Vanderwende. 2008. Using contextual speller techniques and language modeling for ESL error correction. In *Proceedings of IJCNLP 2008*, Hyderabad, India, January.

S. Gui and H. Yang, 2003. *Zhongguo Xuexizhe Yingyu Yuliaoku. (Chinese Learner English Corpus)*. Shanghai Waiyu Jiaoyu Chubanshe, Shanghai. (In Chinese).

Jia-Yan Jian, Yu-Chia Chang, and Jason S. Chang. 2004. TANGO: bilingual collocational concordancer. In *Proceedings of the ACL 2004*, pages 19–23, Barcelona, Spain.

D. Lonsdale and D. Strong-Krause. 2003. Automated rating of ESL essays. In *Proceedings of the NAACL-HLT 03 workshop on Building educational applications using natural language processing*, pages 61–67, Edmonton, Canada.

G. Minnen, F. Bond, and A. Copestake. 2000. Memory-based learning for article generation. In *Proceedings of the Fourth Conference on Computational Natural Language Learning and of the Second Learning Language in Logic Workshop*, pages 43–48.

E. Tjong Kim Sang and S. Buckholz. 2000. Introduction to the conll-2000 shared task: Chunking. In *Proceedings of CoNLL-2000 and LLL-2000*, pages 127–132, Lisbon, Portugal.

D. Schneider and K. F. McCoy. 1998. Recognizing syntactic errors in the writing of second language learners. In *Proceedings of the 17th international conference on Computational linguistics*, pages 1198–1204, Montreal, Quebec, Canada.

C.-C. Shei and H. Pain. 2000. An esl writer's collocational aid. *Computer Assisted Language Learning*, 13(2):167–182.