

A Grammar and Style Checker Based on Internet Searches

Joaquim Moré, Salvador Climent, Antoni Oliver

Open University of Catalonia (UOC)

jmore@uoc.edu

scliment@uoc.edu

aoliverg@uoc.edu

Abstract

In this paper we present an English grammar and style checker for non-native English speakers. The main characteristic of this checker is the use of an Internet search engine. As the number of web pages written in English is immense, the system hypothesizes that a piece of text not found on the Web is probably badly written. The system also hypothesizes that the Web will provide examples of how the content of the text segment can be expressed in a grammatical and idiomatic way. So, after the checker warns the user about the odd character of a text segment, the Internet engine searches for contexts that will be helpful for the user to decide whether he/she corrects the segment or not. By means of a search engine, the checker also suggests the writer to use expressions which are more frequent on the Web other than the expression he/she actually wrote. Although the system is currently being developed for teachers of the Open University of Catalonia, the checker can also be useful for second-language learners, translators, and post-editors.

1. Introduction

The grammar and style checker we present here is currently being developed for teachers of the Open University of Catalonia. The papers these teachers want to publish in journals or international conference proceedings must often be written in English, which is not their mother tongue. Although their command of the language is generally good, most of them do not feel confident enough about the correctness and the idiomatic flavour of their writing. They feel secure about the correctness of a piece of text when they find it in a document already written in English (provided that this document is judged as grammatically and stylistically correct). If the piece of text is not found, the inference that it is probably badly-written is only justified if the number of documents available is very large and the documents are varied. Internet provides an immense number of varied documents written in English; so the main characteristic of our checker is the use of an Internet search engine that detects the text segments that are not found on any web page. For each of these segments, the checker informs the user that the segment is 'brand-new' in the Internet universe and that it may be badly written, which is highly probable when the writer is not a native English speaker and does not have a sound knowledge of the language. Then the checker searches for web pages containing different ways of expressing the content of the segment (variants). From the search results page, contexts with the variants are displayed to the user.

In the Natural Language Generation field, evidence from corpora has been used to choose a particular sentence realization (Langkilde & Knight, 1998; Langkilde, 2002) and Internet search engines have been used for testing error-detection rules in grammar checkers (Naber, 2003). Our corpus-based checker never tells the user how to write. It would be against the creative use of language to judge a segment as 'incorrect' because it is not found on the Web. So the checker just warns the writer and displays the excerpts of the web pages found that contain variants of the segment written. These excerpts are considered useful for the user to detect grammatical and stylistic mistakes, or to decide whether to reword the text or not. Of course, the user can leave the

segment as it is when the examples are not convincing enough for him/her to change it.

2. Description of the Components

The checker has the following components

- User Interface
- Tagger
- Chunker
- Internet search engines
- Brand-new segments detector
- Improvable segments detector
- Searcher and displayer of examples

User Interface

The user interface loads the document the user wants to check (up-to-now the document must be in .txt format). The user can check a particular piece of text by clicking on it. In this case, the system checks the segment selected. If not, the system checks the whole text.

Tagger

The tagger POS tags any string of words. The tagger used has been the demo version of the TreeTagger (Schmid, 1994) for Windows¹. The demo version cannot annotate more than 200 words. Anyway, we have focused on the checking of segments selected by the user, so the number of words hardly ever will surpass this limit. The output of the tagger is a list of tagged words with the following format: Word-POS-Lemma.

Chunker

The chunker splits a POS-tagged piece of text into chunks. The chunks established so far are the following:

- *Nominal*: string of words that are determiners, adjectives or nouns, and form an NP (e.g. *an Internet search engine*)
- *Verbal*: string of words that form single verbs and complex verbs

¹ <http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTreeTagger.html>

- *Verbal+Nominal*: string of words containing a verbal followed by a nominal (e.g. *organise the academic activity*).
- *Nominal+Prep+Nominal*: string of words containing two nominals linked by a preposition (e.g. *laborer on a farm*)
- *Verbal+Prep+Nominal*: string of words containing a verbal and a nominal linked by a preposition (e.g. *carry out a project*)
- *Prep+Nominal*: string of words containing a preposition followed by a nominal. This string is not embedded in a larger chunk (e.g. *on the one hand*)
- *Adverbial+verb+adjective*: string of words containing an adverb and a verb or adjective (e.g. *also display examples*).

The chunks reveal concepts and relationships between the concepts worded in the segment. We consider prepositions and verbs as words that relate concepts.

Search engines

The checker uses the engine of the online Wordnet 2.0² to get lexical information about how concepts can be expressed. The engines used to find the search results for a text segment are the search engine of Yahoo³ and Altavista⁴.

Brand-new/improvable segment detectors

From the search results page, these detectors discern if the segment is brand-new (no exact match found on any web page). If not, the detectors also judge if the segment can be improved (*improvable*).

Searcher and display of examples

When a segment is brand-new or is judged as improvable, this component searches for web pages containing variants of this segment and displays the snippets from the results page. These snippets may be useful for the user to reword the content of the segment. The maximum number of snippets that can be displayed on a search results page has been set to 100.

3. Detection of brand-new and improvable segments

Brand-new segments are those whose search results pages contain the sequence '*We didn't find any Web pages*' or there are no snippets (out of 100) where the exact match is highlighted. The detection of improvable segments is more complex.

3.1 Wordnet and the detection of improvable segments

The improvable segment detector activates the Wordnet search engine in order to find better wordings for a piece of text. For instance, when the syntactic chunk of the text segment is *Prep+Nominal*, the detector hypothesizes that the piece of text is a way of expressing a

concept, or it is a discourse connector. Because of the organization of Wordnet, the engine searches for the synsets of the nominal head. Each synset gloss containing the head in the results page is tagged and split into syntactic chunks. Then the chunk of the text segment is compared to the chunks that contain the head in the glosses. If the chunks coincide except for one non-functional word, then the text segment is regarded as improvable. Let's see an example. Imagine the user wrote

- (1) In the one hand, we explain the antecedents in the study of the cognitive processes...

In the one hand is not brand-new. But the Wordnet engine finds *on the one hand...*, *but on the other hand...* in the gloss for sense 7 of 'hand'. After tagging and chunking the gloss, the detector notices that *on the one hand* forms the same syntactic chunk as *in the one hand*, which does not appear anywhere in the results page. So, the checker displays the following message:

- (2) **hand** -- (one of two sides of an issue; "on the one hand..., but on the other hand...")

This message is the complete Wordnet information for sense 7 of *hand*. This message may be useful for the user to notice that *in the one hand* should be revised.

3.2 Taking advantage of the 'did you mean?'

When the question 'Did you mean...?' appears in the search results page, the guessed form is tagged and split into chunks in order to check if the syntactic structure of the guessed form is the same as the one of the text segment. If so, the guessed form is searched and the number of results is compared to the number of results of the text segment. The text segment is regarded as improvable when the number of its results is smaller. For example, imagine the user wrote

- (3) .. it displays real-English examples with an Internet searcher.

The results page for 'Internet searcher' contains the question *Did you mean 'Internet search'?* *Internet search* is tagged and is identified as a noun phrase, as it was *Internet searcher*. So, the results of *Internet searcher* (1,660) and *Internet search* (3,220,000) are compared. According to the comparison, *Internet searcher* is regarded as improvable.

3.3 Detecting the most frequent variant

A variant of a segment can be a string with the same words but in a different order. See, for example,

- (4) ... in order to detect odd pieces of text and to also display helpful contexts.

If the user wants to check *and to also display*, the adverbial *also* is placed leftmost and then new queries are performed by moving the adverb one position each time from left-to-right. Unfortunately, the results of Yahoo do not vary significantly according to the position of the adverb so we use Altavista for this kind of search. The engine searches for each variant and the detector

² <http://www.cogsci.princeton.edu/~wn>

³ <http://www.yahoo.com>

⁴ <http://www.altavista.com>

compares the number of results (*also and to display: 0, and also to display: 340, and to also display: 13, and to display also: 2*). As the results of ‘and to also display’ only surpass ‘also and to display’, this segment is considered improvable.

4. Displaying helpful contexts

When a segment is considered improvable, the checker displays short excerpts of the web pages that contain the preferred variant. These contexts are the snippets of the results page. The variant appears in boldface type. So, in the case of *Internet search*, the system will display contexts like (5i) and (5ii) .

- (5) i) ...**Internet Search** Tools. Single SearchEngines/Portals ...
- ii) With billions of pages on the Web, you use a search engine if you're looking for something specific. Learn how search engines acquire, store and organize all that data to help you find what you're... ... like most people, you visit an **Internet search** engine.

After reading (5ii) the user who wrote *Internet searcher* may prefer to write *Internet search engine*. This is an example of how the system can be useful for translators, who must deal with terminology.

In the case of *to also display* contexts like (6) will be displayed

- (6) ... Sometimes the use of a spreadsheet can help the pupils to perform calculations more easily and **also to display** their results graphically in the form of bar charts and pie charts. This facility to

As for brand-new segments, the search for helpful contexts is performed by substituting the words that relate terms for a new element. When the segment is a verbal+nominal chunk, the verb is substituted by one of its synonyms. The synonym belongs to the synsets of the verb according to the results page of the Wordnet engine. Then the Yahoo engine searches for documents with the new keywords. If contexts are found, they are displayed to the user. For example, if the user writes the brand-new segment...*to devise the academic activity, devise* is substituted by a different Wordnet synonym (organise, organize, machinate...) in *n* searches where *n* is the number of elements in the verb's synsets. Then, contexts like (7) are displayed to the user.

- (7) Committees including the important General/Professorial/Academic Board, and the Finance Committee ... and lectureships, and **organise the academic activity** of specific departments or ... sub-

If the synonym substitution fails or the brand-new segment does not contain a verb, the words that relate concepts (e.g. prepositions) are substituted by a special symbol that matches any word between the terms related. The system displays the snippets from the results page where the terms are related by a string of words in

boldface form (with no punctuation in between). In this string, the user can see a different preposition other than the one he/she used or learn an idiomatic way of relating the terms. The snippets are tagged and chunked in order to present first the contexts where the boldface words form the same syntactic chunk as in the original text segment. For example, if the user wrote *we carried up a project that lasted 2 years*, where *carried up a project* is brand-new, the checker first displays contexts like HOW WE **CARRIED OUT OUR PROJECT** that may be useful for the user to realize that the preposition should have been ‘out’.

We are thinking of displaying contexts where certain terms of the original text that coexist in the sentence level (with no punctuation in-between) coexist in a more frequently used syntactic chunk. More idiomatic ways of saying the same thing would be presented to the user. For example, it would display **search results page** (an NP with 515,000 results) in case the user wrote *the page that shows the results of the search* (1 result). The system should consider this complex NP as a shorter way of stating the concept relations expressed in the sentence.

5. Comparison with other checkers

The checker we present here is different from the traditional ones in that it is not based on pre-defined language-dependent rules (Naber, 2003), tree-parsings (Jensen et al, 1993) nor statistics (Attwell, 1987). Except for the tagger, the other modules actuate by means of a search engine, which is ‘non-language-dependent’. So the checker would be easy to be adapted to another language provided a tagger for this language exists and can be called by the checker and the number of web pages in this language is huge. On the other hand, we think that this checker can warn the user about a wider scope of phenomena beyond the subject-verb agreement and other typical errors that are dealt with by the traditional systems. Actually, this checker is being developed as a complement of these systems. The spelling and the typical grammar mistakes are already detected by the traditional checkers, so we want to present quite a simple way of assisting a user whose writing reveals aspects difficult to be detected by pre-defined rules.

As the system is currently being developed, we do not have evaluation data about its performance; so a comparison with other checkers has not been carried out yet.

6. Future work

The first thing we want to do is to evaluate how the checker overcomes some problems which are inherent in web-searching. For example, badly-written pages are not discriminated on the Web so the checker does not know for certain if a non brand-new segment matches the mistake of a non-native English writer. The case-insensitive matching also causes some badly written segments to be considered as non brand-new. According to Naber (2003), Google finds the ungrammatical segment ‘the is’ because it matches a document containing ‘*About the IS associates*’, where IS is probably an acronym.

Ungrammatical non brand-new segments are expected to be infrequent on the Web, but what is the minimum number of results necessary to judge a segment as grammatically correct? When the coexisting terms are

very frequent, the threshold can be high (e.g. 'machine translation', 280,000 results) but the presence of a less frequent combination in a perfectly written segment drops the number of results dramatically (e.g. 'machine translation methods', 109 results); so the level should be stated accordingly. We are thinking of applying statistical methods to state the results threshold although other complementary methods are being considered, such as the identification of the reliable urls of the contexts displayed. For example, the documents from urls with .edu or containing 'www.citeseer', the huge on-line library of scientific publications, probably are written in an acceptable English.

Another problem inherent with search engines is their lack of linguistic criteria when matching. For instance, they do not match 'I loved the woman' with documents containing 'I love the women'. We expect that consults to Wordnet, and also the tagging and chunking of snippets can attenuate these effects. This will be analysed and quantified in the near future.

7. References

- Atwell, E., and Elliot, S. (1987) Dealing with ill-formed English text. In *The Computational analysis of English*, Longman.
- Jensen, K., Heidron, G.E. and Richardson, S.D. (Eds). (1993). *Natural language processing: the PLNP approach*. Kluwer Academic Publishers.
- Langkilde, I. & Knight, K. (1998). Generation that exploits corpus-based statistical knowledge. In *Proc. COLING-ACL*.
- Langkilde, I. (2002). An Empirical Verification of Coverage and Correctness for a General-Purpose Sentence Generator. In "Proceedings of the International Language Generation Conference 2002", New York (pp. 17--24).
- Naber, D. (2003). *A Rule-Based Style and Grammar Checker*. Diplomarbeit. Technische Fakultät Universität Bielefeld.
- Schmid, H. (1994). Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proc. of the First International Conference on New Methods in Natural Language Processing (NemLap-94)*, Manchester, England (pp. 44-49).