Avalingua: Natural Language Processing for Automatic Error Detection

Pablo Gamallo, Marcos Garcia

Centro Singular de Investigação em Tecnologias da Informação (CITIUS)

University of Santiago de Compostela

Iria del Río, Isaac González

Cilenis S.L.

Abstract

The objective of this article is to present an automatic tool for detecting and classifying grammatical errors in written language as well as to describe the evaluation protocol we have carried out to measure its performance on learner corpora. The tool was designed to detect and analyse the linguistic errors found in text essays, assess the writing proficiency, and propose solutions with the aim of improving the linguistic skills of students. It makes use of natural language processing and knowledge-rich linguistic resources. So far, the tool has been implemented for the Galician language. The system

has been evaluated on two learner corpora reaching 91% precision and 65% recall (76% F-score) for the task of detecting different types of grammatical errors, including spelling, lexical and syntactic ones.

Keywords: Automated Error Detection, Learner Corpora, Natural Language Processing, Syntactic Analysis, Language Assessment

1. Introduction

This article first describes a linguistic tool for error detection, called *Avalingua*, based on Natural Language Processing techniques and its evaluation on two learner corpora to evaluate its precision and recall.

Avalingua is a linguistic software aimed at automatically identifying and classifying spelling, lexical, and syntactic errors in written language. This tool has been designed to detect and analyse diverse types of linguistic errors, assess writing proficiency, and propose solutions with the aim of improving the linguistic skills of students. It makes use of natural language processing and knowledge-rich linguistic resources. It has been developed, so far, for the Galician language, and can be applied to both L1 learners of Galician (i.e., to investigate language acquisition in children) and L2

language learning.

To evaluate the performance of *Avalingua*, we made use of two learner corpora. The first one is a collection of writing texts belonging to Galician children in the third year of secondary school. The second one consists of texts written by adult Portuguese L2 learners of Galician language. As will be shown later in Section 4, *Avalingua* achieves 91% precision and 65% recall at the task of detecting and classifying different grammatical errors on these two learner corpora. Error correction is out of the scope of the evaluation. The high quality performance achieved in the experimental tests shows that the system is a useful tool that can help teachers assess writing proficiency of students, not only for L1 acquisition, but also for L2 learning.

Moreover, it will be possible to adapt *Avalingua* to other languages than Galician, since the system is based on a modular architecture that sharply separates computational processes from linguistic resources. Given this modular structure, *Avalingua* will be implemented to English and other languages without high computational cost. In the last two years, there has been an increased interest in developing automatic tools for identifying and correcting both spelling and grammatical errors in texts written by English learners (Leacock 2010; Dale & Kilgarriff 2011). This growing interest has led researchers to organize two international competitions, namely HOO-

2011/2012[1] and CoNLL-2013/2014[2] shared task, aimed at comparing the efficiency of different systems trained on a large collection of texts written by learners of English. Those learner corpora are completely annotated with error tags and corrections, and all annotations have been performed by professional English instructors.

This article is organized as follows. We start by introducing in the next section the state of the art in grammatical error detection. Then, Section 3 describes our linguistic tool in terms of objectives, motivations, and internal architecture of the system. In Section 4, we evaluate the performance of the system for Galician language using two learner corpora. Finally, Section 5 addresses some conclusions and an outlook to future research.

2. Automatic Error Detection and Correction

2.1 Previous Research

It has been estimated that over a billion of people are learning second or third languages (Leacock et al. 2010), and the numbers are growing with the

1http://clt.mq.edu.au/research/projects/hoo/
2 http://www.comp.nus.edu.sg/~nlp/conll14st.html

increasing interconnectedness among different populations and cultures. These language learners can provide a huge amount of written text for tools that help identify and correct a great variety of writing errors. Unfortunately, most of these errors are not detected by commercial proofreading tools, since the error distribution is sparse and skewed (i.e., there are many different types of errors occurring few times). In consequence, developers must provide the automatic tools with deep linguistic knowledge to automatically identify and recognise them. To handle this situation, researchers in Natural Language Processing (NLP) have developed linguistic systems that automatically detect and correct grammatical errors made by learners in written texts. The target errors detected by these tools mainly involve the use of determiners (Han et al. 2006), prepositions (Tetreault & Chodorow 2008), both determiners and prepositions (Gamon 2010), and collocations (Dahlmeier & Ng 2011).

Moreover, as it was mentioned in the Introduction, this research has gained popularity recently and the increasing interest in it has resulted in two recent Shared Tasks to detect and correct English texts written by non-native speakers of English: 1) The "Helping Our Own"-task (HOO-2011-2012) (Dale & Kilgarriff 2010, Dale & Kilgarriff 2011, Dale et al. 2012), which was focused on error detection and correction of determiners and prepositions, and 2) the "Conference on Computational Natural Language

Learning" *(*CoNLL-2013-2014) (Ng et al. 2013). Instead of focusing on only determiner and preposition errors as in HOO-2012, the CoNLL-2013 shared tasks include a more comprehensive list of five error types: determiners, prepositions, noun number, verb forms, and subject-verb agreement. In the CoNLL-2014 shared tasks, the list of errors is still larger. Most systems participating in these shared tasks are based on supervised machine learning strategies. They learn a linguistic model from a sample of annotated errors found in the training corpus, and use this model to identify and classify further errors found in new text. However, the performance of the systems participating in these shared tasks is rather low. For instance, the best system in CoNLL-2013 (out of 17 participants) merely achieved an F-score (i.e., the mean proportion between precision and recall) of 42%. Low performance in error detection is due to two problems: first, the inherent difficulty of automatically recognizing sparse instances of error types with skewed distributions in written texts. Second, most machine learning systems are not provided with specific linguistic knowledge and rich external language resources, which can help in the detection and classification process. To minimize these drawbacks, our system, *Avalingua*, relies on rich-knowledge modules and resources, i.e., modules that contain specific grammar rules to detect typical syntactic errors made by the learners, as well as large lexical resources used to identify different

types of lexical problems.

Even if the dominance of rule-based approaches to grammatical analysis gave way in the 1990s to statistical and machine learning methods, the best publicly available grammar checking programs (for instance, AbiWord[3]) are based on rule-based grammar analysis, and not on machine learning techniques (Leacock et al. 2010). It follows that automatic detection of grammar errors is still dominated by rule-based approaches, at least, in the case of the best grammar checking systems.

In rule-based approaches to grammar error detection, two different traditions are found: first, those strategies focused on identifying errors by means of specific *error rules*, which can be defined using pattern matching, robust "ATN parsers" (Liou 1991), which are based on finite state machines, or "Mal-rules", which serve to relate the erroneous input to well-formed semantic representations (Bender et al. 2004). However, the main disadvantages of a strategy based on error rules are the following: 1) it cannot handle unpredictable errors; 2) even if it may reach good precision in detecting the target errors, its coverage tends to be low; 3) as more and more types of errors need to be handled, the grammars or patterns it defines become increasingly complicated.

The second tradition in grammar error detection relies on "Constraint Relaxation" to give a parser the elasticity it needs to perform correct error analysis (Vandeventer 2001). Constraint relaxation produces an environment

3 See http://www.abisource.com/ (27 June 2014)

where relatively “ungrammatical” sentences can be successfully parsed and, then, no new rules are needed to be added to the grammar. A correction can thus be easily generated by examining the violated constraints. However, the major drawbacks of this approach are the following: 1) as deep parsing is required, the analysis is not robust, i.e. the system skips those sentences that it is not able to fully analyze; 2) as it over-generates parses, it results in poor computational efficiency, i.e., in some cases the system does much more than what it is required; 3) it achieves low precision even if it potentially covers any type of error.

Our proposal follows the first tradition to detect grammar errors, since it relies on writing specific error rules. The main advantage of using this type of strategy is that it gives rise to robust and computationally efficient systems, which achieve high precision for the specific errors they target. The main difference between our proposal and other similar systems is that the shallow syntactic parser of *Avalingua* is based on a formal *dependency grammar* provided with a language formalism used by linguists to write error rules.

2.2 Applications

Automatic error detection can be applied to various scenarios. The most popular application of error detection is in proofreading tools such as spell

and grammar checkers, which are used in text processors. However, there is an increasing interest in applying error detection strategies in educational contexts, too, where it has been used for student assessment (Yannakoudakis et al. 2011) and language learning assistance (Chodorow et al. 2010). Two specific applications are found in the field of education: automatically scoring essays and language learning assistance. As it will be described in the following section, *Avalingua* is suited to, not only proofreading, but also to both help teachers scoring essays and to assist language learning.

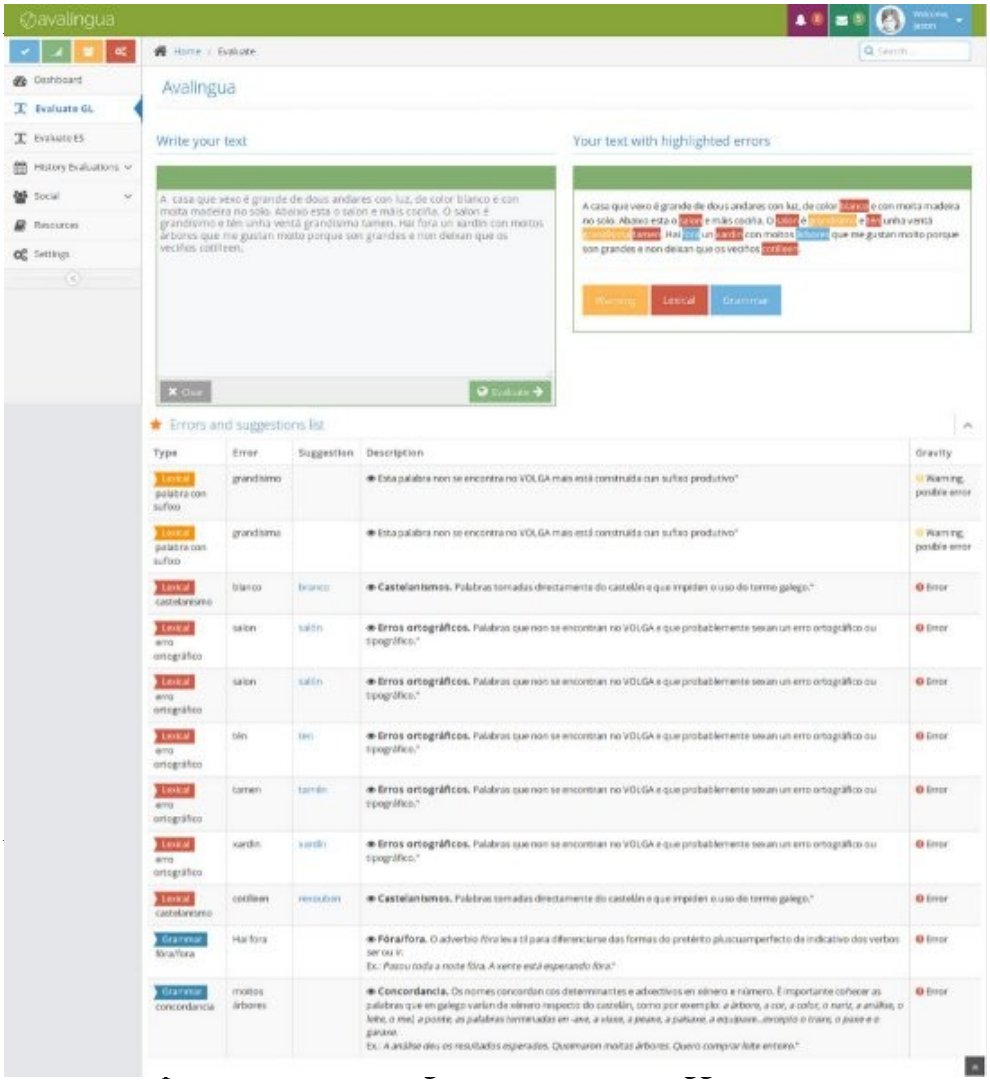
3. *Avalingua*

Avalingua is a software that assesses the correctness of written texts on the basis of a deep, automatic linguistic analysis. The analysis consists of several steps, namely detecting, classifying, and extracting different types of errors by making use of both NLP tools and diverse language resources. Error detection is made at different linguistic levels: spelling, lexis, and syntax. The system also gives a holistic score of the text quality on the basis of the errors that have been detected. Other stylistic elements such as vocabulary distribution or ratio of punctuation marks are considered in order

to identify and filter out non-natural essays randomly generated.

3.1 Target

Figure 1. The front end of *Avalingua*



correction, informs about the gravity of the error, and compares the current text score with the score average computed from the results of other students, allowing to assign language levels like A1, A2, B1, B2, C1, and C2, which are the six levels defined by the *Common European Framework of Reference for Languages* (CEFR; Council of Europe 2009). *Avalingua* can be used, not only to improve writing quality (as traditional language checkers), but also to allow learners to observe their errors and then to

understand the meta-linguistic properties of such errors (Chodorow et al. 2010). This way, the learning assistance can track what the learners are doing incorrectly in order to generate quantitative and qualitative data about their linguistic behaviour, and thereby improve the learning process.

3) Teaching support**:** *Avalingua* can also be used as an automatic assessment method to help teachers or educational administrations to know the language proficiency levels of a single class or a language schoolo. Ware and Warschauer (2006) point out that "[a]nother interesting advantage of this form of electronic feedback, however, includes the large database, of student writing that computers can store". Given a group of students, the system allows the teacher to compute the current linguistic level of the group, the most frequent error types the students make, as well as to follow the development of their learning process. It follows that *Avalingua* allows for the monitoring of both the drawbacks and strengths of an entire student class. This functionality of our system is related to those approaches focused on automatically scoring essays. Error detection is a key component in many systems which automatically provide essay scoring (Yannakoudakis et al. 2011). These systems rely on aspects of grammar and lexical use in order to generate a holistic score measuring the overall text quality.

3.2 Motivations

In spite of the amount of research done, the relation between corrective feedback and learning is still not clear. Truscott (1996) argues against corrective feedback, according to the experiments performed, while Ferris (1999) argues in favour of it, by claiming that Truscott has not posit a clear definition of the term "error correction". Moreover, Ferris claims that most L2 students value corrections and feedback. However, although research has not yet been able to prove which type of feedback is more effective, more arguments have been made in favour of the usefulness of process including corrective feedback: "Although it is unlikely that feedback alone is responsible for long-term language improvement, it is almost certainly a highly significant factor" (Hyland & Hyland 2006). In fact, corrective feedback has generally been found to be beneficial and/or helpful to L2 learning, and most recent studies found positive and significant effects of written corrective feedback (e.g. Russell & Spanda 2006). Some exceptions against the benefit of using corrective feedback can be found in (Truscott & Hsu 2008, Liu 2008, Hartshorn et al. 2010).

Considering that written corrective feedback can help the process of learning, it is possible to enumerate a list of motivations to use an automatic tool such as *Avalingua* for both self-learning and teaching. The main motivations for the use of *Avalingua* (or similar software) in self-learning

are the following:

- Learners can view quickly the grammatical and lexical errors they make without waiting for the detailed correction of a human assistant or the teacher and start to internalize this feedback directly.
- Learners become aware of the most frequent error type(s) they make and are thereby able to find efficient strategies for self-correction.
- Learners acquire maturity and greater autonomy since they take responsibility for the correctness of their own writing. Self-learning helps students to better understand the errors they make, which will result in a clear improvement in their language performance.
- Learners progress at a pace suited to their needs and abilities.
- There is an extra motivation for the learners, since the process of learning spelling, vocabulary, and grammar starts from their own writing.
- The learning process is empirical, since it begins with practice and personal experience before reaching the theory and understanding of generic rules.

As far as teaching is concerned, the educational motivations to use software such as *Avalingua* are the following:

- Essay correction is one of the most time-consuming tasks for

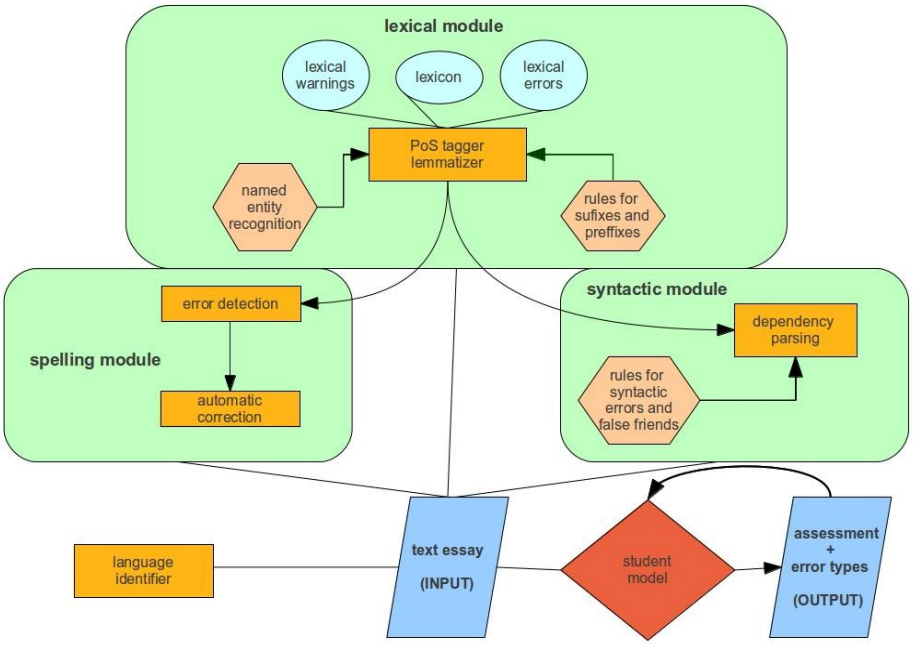
language teachers. *Avalingua* can minimize correction time if the goal is the assessment of purely linguistic performance. Thus, it is possible to make writing one of the main school- and extracurricular activities without significantly increasing the workload of the teachers. By using a system like *Avalingua*, "instructors can free time to turn their attention to other aspects of teaching in the process writing approach" (Ware & Warschauer 2006: 112).

- The system lets teachers relinquish some control and allows the learners to make their own decisions about revising their texts with the support of automatic feedback (Hyland & Hyland 2006).
- The teacher can have a very rough idea of the proficiency level of their students at any time quickly.
- The teacher can effortlessly know the types of the most common and frequent errors made by their students, in order to spend more time to prepare suitable material and building activities.
- Teachers can organize the course by defining a protocol with periodic deliveries of essays, which allows them to better monitor the linguistic progress of their students.
- In the context of a public or private school, it is possible to assess the language level of different classes and courses at any time, and have a clear idea of those groups that require more language reinforcement.

3.3 The system

The linguistic tool consists of several modules and components, most of them relying on NLP techniques such as Part-of-Speech tagging, syntactic parsing, named entity recognition, and language identification. The whole architecture is depicted in Figure 2.

Figure 2. The system architecture of *Avalingua.*



First, we can distinguish between purely linguistic modules and a separated

component we call *student model* (diamond-shaped box in Figure 2). The linguistic modules detect errors in the written essay (the input), while the student model, on the basis of these errors, produces as output a global score of the essay and more accurate assessments at different linguistic levels. The student model also compares the current score with previous assessments made by the student or by the group he/she belongs, and finally updates the model.

The linguistic component is organized in three main language modules: spelling, lexis, and syntax, even if such a distinction is quite artificial in some cases (e.g. with false friends and collocations). In addition, there is a language identifier which has been built as a separate linguistic module. The modules are also interconnected since some tasks of one module rely on the results generated from other modules. For instance, both spell checking and syntactic analysis require information from the lexical module (the central component in Figure 2).

The Linguistic modules consist of resources (small circles in Figure 2), tools (rectangle boxes), and complementary rules or operations (pentagons).

3.3.1 Lexical Module

This is the central component of the system, since the other two linguistic

modules rely on the lexical and morpho-syntactic information it generates. By using several lexical resources and a set of tools, this module generates both the restricted list of word forms that are checked by the speller, and the PoS-tagged lemmas which are syntactically analysed. In addition, it detects lexical errors and warnings. In the following, we describe the resources and tools constituting the module.

3.3.1.1 Resources. As shown in Figure 1, there are three lexical databases (small circles): 1) a standard lexicon of word forms, lemmas, and PoS-tags, which represents the general vocabulary of the target language, 2) a list of frequent lexical errors, and 3) a list of Out-Of-Vocabulary words (OOV) that cannot be perceived as lexical errors but just as lexical warnings (e.g., neologisms, accepted slang and abbreviations).

3.3.1.2 Lexical errors. The lexical errors are those OOVs whose correction cannot be easily suggested by traditional spell checkers since the odd form and the correct word are written with very different strings of characters (i.e. there is a high edit distance[4] between the two strings). Many lexical errors are language interferences between first and second languages. An interference is the transfer of elements of one language into another. So,

4Edit distance between two strings of characters is the number of operations on characters (delete, insert, replace, or transpose) required to transform a string into another.

misspellings caused by interferences can considerably modify the string of the correct word form. For instance, English learners with French as their first language can write **garantie* instead of *guarantee* (Edit distance 2). Those with Spanish as mother tongue could use **bibliotec* instead of *library*, since the Spanish word *biblioteca* means *library*. In this case, the interference gives rise to a completely different word form, which is impossible to detect for a standard spell checker. If the system is used to improve first language learning of native students at primary or secondary school, we can observe other types of lexical errors, namely slang or abbreviations that are not allowed or accepted in formal language but are common in short messages: e.g., *b4* instead of *before*, *bc* instead of *because*, etc.

3.3.1.3 Lexical warnings. Lexical warnings are OOV word forms that must be accepted according to the linguistic criteria defined in the evaluation. Among the different types of warnings, we consider neologisms, domain-specific terminology, frequent Latinisms, compound words, allowed abbreviations, and accepted slang. Notice that neologisms, abbreviations, or slang words can be accepted or not according to different evaluation criteria, for instance the language register required can be distinguished between formal, informal, etc. It is possible for teachers or institutions to modify the

list of errors and warnings according to their own linguistic criteria and learning objectives.

3.3.1.4 Extraction of errors and warnings. One of the most effective strategies to semi-automatically acquire errors and warnings is to extract the most frequent OOV from large corpora, rank them by frequency and classify them. Errors are most easily found in learner corpora as well as in large sets of short messages (e.g. SMS or tweets). As our learner corpora is very small, we have done the extraction of errors from a collection of tweets compiled using a streaming API of Twitter and filtering by language[5]. To find warnings, the extraction can be performed on technical corpora containing texts from diverse knowledge domains. We have performed the extraction on the Galician Wikipedia[6]. All instances of both errors and warnings extracted in such a way are then stored in CSV files, whose structure is very easy to manipulate by teachers and students. As it is shown in Table 1, the structure of a CSV file with lexical errors consists of three columns: the first one contains the error form, the second one the correction and the third one a code representing the error type (slang, abbreviation, Spanish interference, French interference, and so on). The example shown in the table does not correspond with real data, since *Avalingua* has not been

5https://dev.twitter.com/docs/api/streaming
6http://gl.wikipedia.org/

implemented for English yet. This is just an example with the aim of illustrating the internal structure of the CSV files used as lexical databases.

Table 1. A sample of lexical errors.

ill-formed word	*correct word*	*error type*
b4	Before	lex05
Bc	Because	lex05
Bibliotec	Library	lex02
Garantie	guarantee	lex01

3.3.1.5 NLP Tools. The main tool of the lexical module is a PoS-tagger which assigns a Part-Of-Speech category and a lemma to each token of the input text. In order to allow *Avalingua* to be applied to diverse languages, we make use of a multilingual tool, "FreeLing" (Padró & Stanilovsky 2012), which works for more than 10 languages. This tool also provides us with a Named Entity Recognizer (NER) which identifies proper names, dates, quantities, and other types of OOV word forms which must not be considered as lexical errors even if they are not found in the standard lexicon of a language. In addition, we also implement a sub-module to identify other OOV word forms that must not be considered as incorrect expressions, since they are well formed using derivative and productive affixes (prefixes and suffixes), for instance, adverbial suffixes as *-mente* (-

ly).

The main function of the lexical module is to detect and classify the lexical errors and warnings found in the text. Additionally, this module processes the input text and builds two different outputs. First, it generates the set of word forms (and their linguistic contexts) which is the input of the spelling module. This set was created by filtering out lexical errors, named entities, OOV words correctly built with affixes, and other OOV warnings. Second, the other output of the lexical module is the PoS-tagged and lemmatized text, which is the input of the syntactic module. All this information is finally processed by the student model in order to generate the final assessment and learning suggestions.

3.3.2 Spelling Module

This module, which is not far from a standard spell checker, consists of two related tasks: error detection and automatic correction.

Error detection takes as input the set of filtered word forms given by the lexical module and, according to the vocabulary of the language, identifies ill-formed forms that are sent to the other task: automatic correction.

The process to automatically suggests that the most likely correction

given a misspelled word relies on the algorithm we have implemented for the TweetNorm Shared Task at SEPLN 2013 (Alegria et al. 2013), where the objective was to detect and correct usual errors in Spanish tweets (tweet normalization). The system we presented to the Shared Task achieved the second best performance out of 13 participants (Gamallo et al. 2013). For each misspelled form, the algorithm generates a list of word candidates that are found in the general vocabulary with an edit distance equal to 1. Then, it ranks the list of candidates by taking into account both contextual information from a language model and internal information from orthographic and morpho-phonetic rules.

The spelling errors detected by this module as well as the corrections generated are finally processed by the student model.

3.3.3 Syntactic Module

This module identifies different types of syntactic errors (and grammatical warnings) by making use of a multilingual syntactic parser, *DepPattern*, which we have designed and implemented. The parser relies on a dependency-based grammar whose formalism was described in Gamallo and González (2011). To detect syntactic errors, the grammar underlying *DepPattern* contains two types of rules: *standard rules* to identify syntactic

dependencies and *error rules* to detect ill-formed relationships between words. Below, sentence (1), which is an invented example, contains a very common case of subject-verb agreement mismatch.

(1) The child immediately go to Heaven.

In order to detect this grammatical problem, our grammar makes use of a specific error rule that finds unmatched number values between two dependent words: the noun *child* in the singular and the verb *go* in the plural. Yet, before applying the error rule, it is necessary to identify that *child* and *apply* are related. This is performed using the standard part of the grammar containing dependency-base rules. So, since only error rules are relevant for the assessment made by *Avalingua*, they are classified in diverse types, as we made for lexical errors: concordance mismatch, incorrect preposition, wrong article, etc. Some of the error rules are pre-classified as warnings when they are used to detect syntactic oddities that are not clear mistakes.

Among the syntactic error types, special attention is given to the identification of both false friends and wrong collocations or idiomatic word combinations. False friends are word pairs where a word in one language has a very similar string to a word in another language (they are cognates),

but they have different meanings. For instance, the Spanish verb *contestar*, which means "to answer" is very similar to the English verb *to contest*. As they have similar strings, Spanish students could use *to constest* with the meaning of *to answer* when writing an English essay. Error rules can be effective for detecting this kind of lexical-syntactic mistakes. More precisely, we can define rules to detect the wrong use of the transitive verb *to contest* within ditransitive constructions or just with *to* + NP-complements (**I contested to him*).

On the other hand, error rules are also useful for the identification of incorrect collocations, which in many cases are also caused by interference of the first language. A very common mistake made by Spanish learners is the use of the incorrect idiomatic expression *ser contento* instead of the correct one *estar contento*, with the meaning "to be happy". As for false friends, it is also possible to define error rules to detect different variations and lingustic contexts containing such an incorrect collocation: e.g. **era muy contento*, (Eng: "he/she was very happy"), **somos hoy muy contentos* (Eng: "we are very happy today"). Notice that both false friends and collocations are complex linguistic phenomena that share lexical and syntactic properties. However, in our system they are situated at the syntactic level since the detection procedure requires deep syntactic information.

3.3.4 Language identification

Avalingua also includes a module that aims at identifying the language in which any part of the input text is written. More precisely, the language identifier finds citations and quotations within the text and detects if they are written in the same language as the rest of the text or not. Thus, the objective of this module is to filter out those phrases or paragraphs that were written in a different language from the main language of the text. We have designed and implemented the language detector, called *QueLingua*,[7] used by *Avalingua.* Even if *QueLingua* identifies 8 different languages so far, *Avalingua* does not use it to detect a particular language, but just to guess if any quotation or citation is written in the target language.

3.3.5 Student model

In the current prototype of *Avalingua* we have implemented a few functionalities of the student model. The input essay is scored in a holistic manner by taking into account the number of errors found in the text and the global total size length of the essay. This module also reports statistical information on the types of errors made by the student: error rate and error

7*QueLingua* is freely available at:
http://gramatica.usc.es/~gamallo/quelingua/QueLingua.htm (27 June 2014).

history tracking. In addition, each error/warning found in the text is associated with a linguistic explanation as well as and a suggested correction. The student model is enriched with the scoring information obtained from each individual assessment. So, it is always updated with the results obtained in the last assessment.

4 System evaluation

The system described in the previous section is has a generic architecture that can be implemented for any language. Any specific implementation requires the following three two language-dependent tasks:

1. To define and codify a typology of linguistic errors and warnings for the target language.
2. To build appropriate databases for lexical errors and warnings (this task can be performed in a semi-automatic way), to write specific error rules for the *DepPattern* grammar.

So far, *Avalingua* has been implemented for the Galician language. The human cost of creating the language-dependent resources by means of the three tasks introduced above varies in function of the quality and quantity of those resources. The resources of the current version of *Avalingua* for the

Galician language (i.e. list of error types, database of lexical errors, and error rules) have been elaborated developed in about three months by two researches: a linguist and a computational linguist. No computer engineer is required for these tasks. This is the cost of building a working version of *Avalingua* to another language.

In the remaining of the section, we will describe the general features of this specific implementation, the experiments that have been performed by making use of learner corpora, and the evaluation protocol.

4.1 A specific implementation

Galician is a Romance language spoken by about 3 million people mainly in Galiza, an Autonomous Community located in north-western Spain. Galician is the official language of the community, along with Spanish, and it is recognized as the first language of the local administrations and regional government. It belongs to the same linguistic family as the Portuguese language, and both share a common origin. The two official languages in the community, Galician and Spanish, are taught bilingually in both primary and secondary education, and most students, even those that have Galician as their mother tongue, make many writing mistakes errors, which are likely to be due to language interferences. Thus, in this context,

the problems underlying L1 Galician acquisition are those that are also found in L2 learning.

We decided to implement *Avalingua* for the Galician language in order to help educational institutions of the community monitor and evaluate systematically the learning process of this language at secondary school level.

For this purpose, an open set of errors and warning types was defined:

- 11 types of lexical errors (e.g., interferences from other languages, false friends, previous spelling norms, etc.)
- 15 types of lexical warnings (e.g., neologisms, Latinisms, derivative out of vocabulary words, etc.).
- 29 types of syntactic errors (e.g., agreement, wrong use of prepositions, wrong position of pronouns, etc.).
- 4 types of syntactic warnings (e.g., syntactic issues that are very complex and difficult to be automatically detected)

In addition, by taking into account these types,, we built the lexical and syntactic resources containing a large number of instances of both lexical and syntactic errors/warnings, classified by their respective types:

- 60,000 classified lexical errors
- 19,000 classified lexical warnings
- 185 specific error/warning rules in the syntactic module

Most instances in the lexical resources were extracted in a semi-automatic way. As many lexical errors are likely due to Spanish interferences, we used electronic bilingual dictionaries Spanish-Galician to extract thousands of instances of that error type. More precisely, when a Spanish entry in the bilingual dictionary was not found in a monolingual Galician lexicon, then an error instance was added to the database: the Spanish entry is the ill-formed word and its Galician translation (or a list of possible translations) is the correct form. For other types of less common errors (archaisms, Portuguese interferences, forms from previous linguistic norms, etc.), we mainly used manually made lists published by teachers, linguists, and educational institutions.

Lexical errors and warnings were expanded to obtain all their possible forms using an automatic conjugator for verbs (Gamallo et al. 2013) and inflectional rules for nouns and adjectives.

We did not make use of a *development corpus* to elaborate the resources and rules required by the system. Lexical resources were built

semi-automatically from other existing resources, and error rules were elaborated using grammar manuals and other didactic materials for the Galician language.

4.2 The learner corpora

In order to evaluate the performance of the current implementation of *Avalingua* for Galician, we made use of two learner corpora. The first one is a collection of 22 text essays (35,910 tokens) written by Galician children in the 3th year of secondary school. The second one consists of 8 texts (5,078 tokens) written by adult Portuguese L2 learners of Galician language. We collected the two corpora thanks to the help of two teachers who provided us with the previously anonymized texts. In sum, the 30 texts contain 40,988 tokens and, then, in average there are 1,366 words per text. All texts are compositions written in Galician language that belong to an academic and controlled context of Galician learning. However, our texts belong to two different learning situations:

- L1 Galician learning of native children and;
- L2 Galician learning of Portuguese adults (undergraduate students).

Despite the clear differences between these two learning corpora, they can be considered as comparable in our approach because the point we are interested in is the automatic evaluation of writing proficiency in a language (in this case, Galician). Writing proficiency in Galician language is defined in our evaluation protocol against the established Galician standard. In addition, the use of texts belonging to these two learning situations lets us include in our research a wide scope of samples from Galician learners with different profiles, with different learning levels, and with (possible) different errors.

4.3 Evaluation protocol

The objective of the evaluation is to measure the system's performance in terms of *precision* and *recall*, by comparing its output with annotator's judgements, i.e. with a gold standard made by human evaluators. The evaluation is just focused on detecting and classifying spelling, lexical, and syntax errors, but not in on correcting them. The detection of non-errors is not considered (Chodorow et al. 2012).

To compute precision, it is necessary to define and identify both true positives (TP) and false positives (FP). TP are the number of correct decisions made by the system while FP are the number of incorrect

decisions. A decision is considered as correct if only if the error is correctly detected and classified. So, to measure the system's performance, we only consider error detection and classification, while error correction is out of the scope (in fact, the current version of our system does not make automatic correction, but just offers suggestions for correction). Given TP and FP, precision is computed as the number of correct decisions made by the system (TP) divided by the total number of decisions made by the system (TP+FP):

To define recall, it is required to identify false negatives (FN), i.e. the number of good decisions that the system does not make. Given TP and FN, recall is computed as the number of correct decisions made by the system (TP) divided by the total number of decisions found in the annotated gold standard (TP+FN):

4.4 Results

Precision was computed by making use of all written essays constituting the

two learner corpora described above. The effort needed to compute this measure is not very high since it just requires a human evaluator to revise and annotate the output of the system. Thus, it is not required to completely annotate all text essays. Table 2 shows the precision results obtained by *Avalingua* from the two test corpora: the text essays written by both Galician native children and Portuguese adults. The output was manually reviewed by a linguist who identified true and false positives.

Table 2: Precision obtained from essays written by Galician secondary school students and Portuguese adults.

	Galician children		**Portuguese adults**		**Precision**
	TP	**FP**	**TP**	**FP**	
lexical errors / warnings	595	23	160	12	.955
syntactic errors / warnings	122	39	18	13	.729
Precision	**.920**		**.876**		**.913**

To show the differences between lexical and syntactic levels, we have separated lexical from syntactic errors/warnings. For the shake of simplicity, lexical errors/warnings also include spelling errors. The results in Table 2 allows us to observe that precision follows the same tendency in the two

learner corpora: high precision in the case of lexical errors/warnings (95%) and lower performance on syntactic ones (73%). On average, precision achieves 91% in the two corpora, namely 92% in the larger corpus (essays of Galician students) and close to 88% in the smaller one (essays of Portuguese adults).

In order to compute recall, we selected a subset of 1,266 word tokens from the test corpus of essays written by the Galician students. Then, two gold standards were built by two different human annotators, in this case two language teachers. For this purpose, the annotators were asked to detect all lexical and grammatical errors in the selected test corpus and classify them (when possible) according to the types defined in *Avalingua*. Results are shown in Table 3.

Table 3: Recall of *Avalingua* obtained from two different Annotators (gold standards).

	Recall (lexical)	**Recall (syntactic)**	**Recall (all)**
Annotator 1	.670	.594	.648
Annotator 2	.657	.643	.651
Average Recall	**.663**	**.619**	**.650**

In this evaluation, the objective of recall is to compare the number of errors/warnings the system correctly detects and classify (TP) to the total

number of errors/warnings found in the annotated gold standards (TP+FN). So, this metric is focused on false negatives, i.e. on those errors/warnings that are found by the annotators and that were not found by the system (FN). The final average recall of *Avalingua* achieves 65%, and there are no significant differences between the two annotators. On the other hand, unlike precision, the differences between lexical and syntactic errors/warnings do not seem crucial (66% against 62%). The harmonic mean of precision and recall, called F-score, is 0.758.

We realized, however, that the manual evaluation performed by the two annotators (who are experienced language teachers) was far from being perfect. The differences between the two annotators were mainly not due, not to a lack of agreement, but rather to a lack of attention, since they were not able to detect all the errors in the test corpus. We found some errors correctly detected by *Avalingua* (when precision was computed) that were not detected by the two evaluators. Taking into account this information, a new meta-evaluation was performed where the gold standard is the union of correct decisions taken from both the output of the system and the two evaluators. We call "Pooling" this new gold standard "pooling". Results are shown in Table 4.

This new evaluation shows that the average recall of the human annotators (language teachers) merely reaches 87%. It follows that human

correction is far from covering all possible linguistic errors of a writing essay. In addition, it also shows that the distance between *Avalingua* and a human annotator in terms of recall is not the difference between 65 and 100% recall (as Table 3 seemed to point out), but just 23 points. i.e., the difference between 65% (average recall in Table 3) and 87% (average recall in Table 4).

Table 4: Recall of the two human annotators against a new gold standard (pooling of all correct decisions).

	Recall (Annotator 1)	**Recall (Annotator 2)**	**Average Recall**
Pooling	.843	.892	**.868**

This new evaluation shows that the average recall of the human annotators (language teachers) merely reaches 87%. It follows that human correction is far from covering all possible linguistic errors of a written essay. In addition, it also shows that the distance between *Avalingua* and a human annotator in terms of recall is not the difference between 65% and 100% recall (as Table 3 seemed to suggest), but just 23 differs in 23 points. i.e.,

the difference between 65% (average recall in Table 3) and 87% (average recall in Table 4).

4.5 Error analysis and discussion

As far as precision is concerned, most incorrect decisions (FP) made by our system come from two sources of errors: wrong detection of syntactic phenomena related to clitics, and odd lexical warnings derived from wrong affix identification. By contrast, spelling and lexical errors are detected with high precision.

Among the wrong syntactic detections, the most frequent error of *Avalingua* is related to the placement of clitics, which is a critical problem in Galician and European Portuguese. In Galician, the natural placement of a clitics is after the verb, but they can also be placed before in some specific contexts. Many failures of the system come about when the clitic turns out to be the very frequent and ambiguous form "*o*" (Eng. "it"), which can also be a determiner (Eng. "the"). Such an ambiguity leads the PoS-tagger to fail in some difficult cases by choosing the incorrect PoS-tag and, thereby, transferring the error to the syntactic parser.

Another source of cause for incorrect decisions comes from the rules used to identify well-formed suffixes and prefixes within OOV words,

which are the most frequent lexical warnings. An example of an incorrect decision is to consider the OOV "*veziños*" a lexical warning, since the system identifies both the diminutive suffix "*-iños*" and the noun "*vez*" (Eng: "turn"). However, this noun has no diminutive meaning in Galician and, it cannot therefore be derived with the suffix "*-iño(s)*". In fact, "*veziños*" is a misspelling of the word "*viciños*" (Eng: "neighbors").

Concerning recall, as it was expected, most false negatives (FN) are false friends that have not been detected and types of syntactic errors that are not treated in the formal grammar. Unlike lexical errors, which are organized in a few types, syntactic errors have a skewed distribution with a long tail constituted by many types with low frequencies. This makes them difficult to detect.

The experiments performed are far from being conclusive since the size of the learner corpora is quite small, in particular the test sample used for computing recall. The main conclusion drawn from this evaluation is that the gap between manual and automatic assessment is still very large in terms of quality and performance. However, if we consider the efficiency underlying automatic linguistic tools, we must point out their ability to assess and score thousands of writing essays in a few seconds. Besides, we are not considering here another relevant factor in human assessment that does not affect automatic tools: after manually correcting and scoring

dozens of essays, tiredness and relaxing concentration can result in low quality assessment.

5 Conclusions

The linguistic tool described in this article, *Avalingua*, has been designed and developed for diverse tasks such as writing assistance, individual learning, and teaching support in writing proficiency assessment. The performance achieved in the experimental tests (91% precision and 65% recall) shows that the system should be improved in order to be a useful tool for the above mentioned tasks, not only in the context of L1 acquisition, but also for L2 learning. The article has provided a description of how the tool has been evaluated against two learner corpora, by taking into account different types of spelling, lexical, and grammatical errors/warnings.

In our current work, we are trying to solve some of the most frequent problems found in the test evaluation. In particular, in order to minimize the syntactic problems inherited from the PoS-tagging process, we are have implemented an intermediate level, between tagging and syntax analysis, which uses specific lexical-syntactic rules to correct PoS-tagging errors. This technique is inspired by the work described in Garcia & Gamallo

(2010). Moreover, we are trying to increase the coverage of the error rules. For this purpose, along the skewed distribution of syntactic errors, we are studying the most frequent types found in the learner corpora, with the aim of implementing more accurate rules for detecting them.

Given the modular structure of the system described in this article, it is possible to adapt *Avalingua* to other languages, such as English, by just creating appropriate resources and formal grammars. There has been an increased interest in developing automatic tools for identifying and correcting grammatical errors in texts written by English learners (Ng et al. 2013). This growing interest has led companies to design high-level grammar checkers, and researchers to organize international competitions (e.g., CoNLL-2013 shared task), aimed at comparing the efficiency of different systems trained on large collections of texts written by English students. Those learner corpora were fully annotated by professional English instructors. Thanks to these freely available annotated corpora, it will become possible to train and design new systems for the English language. In future work, our goal is to adapt *Avalingua* to English texts, mainly by using as source of errors the learner corpora available from the above mentioned competitions as sources of errors. More precisely, we will analyse the more most frequent errors found in those corpora, build lexical resources on the basis of them, and define appropriate correction rules for

our syntactic parser.

References

Alegria, I, Aranberri, N., Fresno, V., Gamallo, P., Padró, Ll., San Vicente, I. Turmo, J. & Zubiaga, A. 2013. Introducción a la tarea compartida Tweet-Norm: Normalización léxica de tuits en español. In *Proceedings of the Tweet Normalisation Workshop at SEPLN-2013*, 38-46. Sociedad Española para el Procesamiento del Lenguaje Natural <http://nlp.lsi.upc.edu/publications/papers/tweetnorm13.pdf> (1 July 2014).

Bender, E. M., Flickinger, D., Oepen, S., Walsh, A. & Baldwin, T. 2004. ARBORETUM: Using a Precision Grammar for Grammar Checking in CALL. In *Proceddings of the InSTIL/ICALL Symposium on Computer Assisted Learning*, Venice, Italy <http://project.cgm.unive.it/events/ICALL2004/papers/020bender.pdf> (1 July 2014).

Chodorow, Martin, Gamon, M. & Tetreault, J. 2010. The utility of article and preposition error correction systems for English language learners: Feedbak and assessment. *Language Testing* 27(3): 419-436 <http://xa.yimg.com/kq/groups/13354653/292314615/name/419.full.pdf>

(1 July 2014).

Chodorow, M., Dickinson, M., Isarel, R. & Tetreault, J. 2012. Problems in evaluating grammatical error detection systems. In *Proceedings of the International Conference on Computational Linguistics (COLING 2012),* 611-628. Martin Kay, Christian Boitet (eds.), Mumbai, India: Association for Computational Linguistics.

Council of Europe. 2009. Relating Language Examinations to the Common European Framework of Reference for Languages: Learning, Teaching, Assessment (CEFR). A Manual. Strasbourg: Language Policy Division <https://biblio.ugent.be/publication/4270320/file/4270333.pdf> (1 July 2014).

Dahlmeier, D. & Tou Ng, H. 2011. Grammatical Error Correction with Alternating Structure Optimization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011)*, 915-923. Portland, Oregon: Association for Computational Linguistics <http://www.aclweb.org/anthology-new/P/P11/P11-1092.pdf> (1 July 2014).

Dale, R., Anisimoff, I. & Narroway, G. 2012. HOO 2012: A report on the preposition and determiner error correction shared task. In *Proceedings of the Seventh Workshop on Innovative Use of NLP for Building Educational Applications*, 54-62. Montréal, Québec, Canada: Association

for Computational Linguistics <http://clt.mq.edu.au/~rdale/publications/papers/2012/BEA2012.pdf> (1 July 2014)

Dale, R. & Kilgarriff, A. 2011. Helping Our Own: The HOO 2011 Pilot Shared Task. In *Proceedings of the 13th European Workshop on Natural Language Generation (NLG'11) at EMNLP 2011.* Belz, A., Evans, R., Gatt, A. and K. Striegnitz (eds.), 242-249. Nancy, France: Association for Computational Linguistics <http://clt.mq.edu.au/research/projects/hoo/hoo2011/reports/HOO2011FinalReport.pdf> (1 July 2014).

Dale, R. & Kilgarriff, A. 2010. Helping Our Own: Text Massaging for Computational Linguistics as a New Shared Task. In *Proceedings of the 6th International Natural Language Generation Conference (NLG'10)*, Kelleher, J.D., Mac Namee, B., van der Sluis, I. (eds.): 263-267 <http://203.144.248.23/ACM.FT/1880000/1873779/p263-dale.pdf> (1 July 2014).

Ferris, D. 1999. The case for grammar correction in L2 writing classes: a response to Truscott (1996). *Journal of Second Language Writing*, 8: 1-11.

Gamallo, P., Garcia, M. & Pichel, J. R. 2013. A method to Lexical Normalisation of Tweets. In *Tweet Normalisation Workshop at SEPLN-2013*, 81-85 <http://gramatica.usc.es/~gamallo/artigos-

web/TWEETNORM2013.pdf> (1 July 2014).

Gamallo, P. & González, I. 2013. A Depurative Strategy for Dependency Parsing with Finite-State Transducers. Submitted to the journal *Linguistic Issues in Language Technology*.

Gamallo P., Garcia, M., González, I., Muñoz. M. & Del Río, I. 2013. Learning verb inflection using Cilenis conjugators. *Eurocall Review* 21(1): 12-19 <http://eurocall.webs.upv.es/documentos/newsletter/download/No21_1.pdf#page=12> (1 July 2014).

Gamallo, P. & González, I. 2011. A Grammatical Formalism Based on Patterns of Part-of-Speech Tags. In *International Journal of Corpus Linguistics* 16(1): 45-71.

Gamon, M. 2010. Using Mostly Native Data to Correct Errors in Learners' Writing: A Meta-Classifier Approach. In *Proceedings of HLT '10 Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2010)*, 163-171. Association for Computational Linguistics.

Garcia, M. & Gamallo, P. 2010. Using Morphosyntactic Post-processing to Improve POS-tagging Accuracy. In *Proceedings of the 9th International Conference on Computational Processing of Portuguese Language*

(PROPOR 2010). Extended Activities Proceedings, Porto Alegre, Brasil <http://gramatica.usc.es/~gamallo/artigos-web/PROPOR2010Web.pdf> (1 July 2014).

Han, N., Chodorow, J. R. & Leacock, C. 2006. Detecting errors in English article usage by non-native speakers. *Natural Language Engineering* 12(2): 115-129.

Hartshorn, K. J. Evans, N. W. Merrill, P. F. Sudweeks, R. R., Strong-Krause, D. & Anderson, N. J. 2010. Effects of dynamic corrective feedback on ESL wiring accuracy. *TESOL Quarterly* 44*:* 84-109.

Hyland, K. & Hyland, F. 2006. State of the art article: Feedback on Second Language students' writing. *Language Teaching* 39, (2): 83-101.

Leacock, C., Chodorow, M., Gamon, M. & Tetreault J. 2010. *Automated Grammatical Error Detection for Language Learners*. San Rafael, CA: Morgan & Claypool Publishers: 1-134.

Liou, H.-C. 1991. Development of an English Grammar Checker: A Progress Report. *CALICO Journal* 9, (1): 57-70.

Liu, Y. 2008. The effects of error feedback in second language writing. *Arizona working papers in SLA & Teaching* 15: 65-79.

Ng, H., Wu, S., Wu, Y., Hadiwinoto, C. & 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 (CoNLL-2013 Shared Task), 1-14. Sofia, Bulgaria: Association for Computational Linguistics <http://www.comp.nus.edu.sg/~nlp/conll14st/conll14st-book.pdf#page=11> (1 July 2014).

Padró, Ll., & Stanilovsky, E. 2012. FreeLing 3.0: Towards Wider Multilinguality. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2012),* Istanbul, Turkey: European Language and Resources Association <http://upcommons.upc.edu/e-prints/bitstream/2117/15986/1/freeling30_Paper.pdf> (1 July 2014).

Russell, J., & Spada, N. 2006. The effectiveness of corrective feedback for the acquisition of L2 grammar. A metaanalysis of the research. In *Synthesizing research on language learning and teaching*, J. M. Norris & L. Ortega (eds.), 133-164. Amsterdam/Philadelphia: John Benjamins

Tetreault, J. & Chodorow, M. 2008. The ups and downs of preposition error detection in ESL writing. In *Proceedings of the International Conference on Computational Linguistics (COLING 2008),* 865-872. Manchester, UK: Association for Computational Linguistics <http://192.5.53.208/u/tetreaul/tetreault-chodorow-coling08.pdf> (1 July 2014).

Truscott, J., & Hsu, A. Y. 2008. Error correction, revision, and learning. *Journal of Second Language Writing* 17: 292–305.

Truscott, J. 1996. The case against grammar correction in L2 writing classes. *Language Learning* 46: 327-369.

Vandeventer, A. 2001. Creating a grammar checker for CALL by constraint relaxation: a feasibility study. *ReCALL* 04/2001: 110-120.

Ware, P. D. & Warschauer, M. 2006. Electronic feedback and second language writing. In *Feedback in Second Language Writing: Contexts and Issues*, K. Hyland & F. Hyland (eds.), 105-122. Cambridge: Cambridge University Press.

Yannakoudakis, H., Briscoe, T., & 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 (ACL 2011)*, 180-189. Portland, Oregon: Association for Computational Linguistics <http://www.aclweb.org/anthology/P/P11/P11-1019.pdf> (1 July 2014).