A Framework for Classification and Practice of Verb Tenses in English Language

Kledilson Endrigo de Souza Ferreira

Supervisor: Álvaro Rodrigues Pereira Júnior
Universidade Federal de Ouro Preto

A Framework for Classification and Practice of Verb Tenses in English Language

Kledilson Endrigo de Souza Ferreira

Supervisor: Álvaro Rodrigues Pereira Júnior
Universidade Federal de Ouro Preto

Dissertação submetida ao
Instituto de Ciências Exatas e Biológicas
Universidade Federal de Ouro Preto
para obtenção do título de Mestre em Ciência da Computação

F383f Ferreira, Kledilson Endrigo de Souza.
A framework for classification and practice of verb tenses in english language [manuscrito] / Kledilson Endrigo de Souza Ferreira. - 2017.
80f.: il.: color; tabs; mapas.

Orientador: Prof. Dr. Álvaro Rodrigues Pereira Júnior.

Dissertação (Mestrado) - Universidade Federal de Ouro Preto. Instituto de Ciências Exatas e Biológicas. Departamento de Computação. Programa de Pós-Graduação em Ciência da Computação.
Área de Concentração: Ciência da Computação.

1. Língua inglesa - Morfologia. 2. Aprendizagem supervisionada (Aprendizado do computador). 3. língua inglesa - Aprendizagem. I. Pereira Júnior, Álvaro Rodrigues. II. Universidade Federal de Ouro Preto. III. Titulo.

CDU: 004

Catalogação: www.sisbin.ufop.br

MINISTÉRIO DA EDUCAÇÃO E DO DESPORTO
Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Biológicas – ICEB
Programa de Pós-Graduação em Ciência da Computação

decom
departamento de computação

Ata da Defesa Pública de Dissertação de Mestrado

Aos 31 dias do mês de janeiro de 2017, às 15:00 horas na Sala de Seminários do DECOM no Instituto de Ciências Exatas e Biológicas (ICEB), reuniram-se os membros da banca examinadora composta pelos professores: **Prof. Dr. Álvaro Rodrigues Pereira Júnior (presidente e orientador), Prof. Dr. Anderson Almeida Ferreira e Prof. Dr. Marco Antônio Pinheiro de Cristo**, aprovada pelo Colegiado do Programa de Pós-Graduação em Ciência da Computação, a fim de arguirem o mestrando **Kledilson Endrigo de Souza Ferreira**, com o título **"A Framework for Classification and Practice of Verb Tenses in English Language"**. Aberta a sessão pelo presidente, coube ao candidato, na forma regimental, expor o tema de sua dissertação, dentro do tempo regulamentar, sendo em seguida questionado pelos membros da banca examinadora, tendo dado as explicações que foram necessárias.

Recomendações da Banca:

(X) Aprovada sem recomendações

( ) Reprovada

( ) Aprovada com recomendações: _____

Banca Examinadora:

_____
Prof. Dr. Álvaro Rodrigues Pereira Júnior

_____
Prof. Dr. Anderson Almeida Ferreira

_____
Prof. Dr. Marco Antônio Pinheiro de Cristo

_____
Prof. Dr. Anderson Almeida Ferreira
Coordenador do Programa de Pós-Graduação em Ciência da Computação
DECOM/ICEB/UFOP

Ouro Preto, 31 de janeiro de 2017.

A Framework for Classification and Practice of Verb Tenses in English Language

Abstract

This work describes a framework developed for verb tenses classification in the English language and automatic generation of verb-tenses-oriented exercises, based on chosen texts for study. The classification is a rule-based process. The rules were analyzed and improved in the course of the work. This research brings a method for generation of exercises of transposition/transformation of verb tenses. By this method, it is possible to transpose each verb tense on a sentence to another verb tense, if there is an indexed example of that target verb tense. The verb tenses transposition method can also be applied in other contexts, like text generation, making it easy to choose an action and applying it to verb tense. Our results demonstrate that the framework is effective for its purpose, reaching 98.17% in average precision on the task of classifying verb tenses.

Um Framework para Classificação e Prática de Tempos Verbais na Língua Inglesa

Resumo

Este trabalho descreve um framework desenvolvido para a classificação de tempos verbais em textos de língua inglesa e geração automática de exercícios focados em tempos verbais, baseados em textos escolhidos para estudo. A classificação é baseada em regras que foram analisadas e melhoradas ao decorrer do trabalho. Esta pesquisa traz um método para geração de exercícios de transposição/transformação de tempos verbais. Através desse método, é possível transpor cada tempo verbal em uma sentença para um outro tempo verbal, se houver exemplo do tempo verbal destino indexado para o processo. O método de tranposição de tempos verbais também pode ser aplicado em outros contextos, facilitando a aplicação de uma ação a um tempo verbal. Nossos resultados demostram que o framework é efetivo para o seu propósito, tendo uma precisão média de 98.17% na tarefa de classificação de tempos verbais.

Declaração

Esta dissertação é resultado de meu próprio trabalho, exceto onde referência explícita é feita ao trabalho de outros, e não foi submetida para outra qualificação nesta nem em outra universidade.

Kledilson Endrigo de Souza Ferreira

Thanks to my family, my friends, my teachers, advisor and everyone who somehow contributed or supported this research.

Contents

List of Figures

List of Tables

Abbreviations

CALL Computer-Assisted Language Learning

FC Future Continuous

FGT Future Going To

FPC Future Perfect Continuous

FPS Future Perfect Simple

FS Future Simple

MALL Mobile-Assisted Language Learning

PaC Past Continuous

PaPC Past Perfect Continuous

PaPS Past Perfect Simple

PaS Past Simple

PeC Present Continuous

PePC Present Perfect Continuous

PePS Present Perfect Simple

PeS Present Simple

List of Algorithms

Chapter 1

Introduction

Learning a foreign language has become a common need for many people in several countries. People do it for jobs, traveling, studying or just for hobby. Personal computer, mobile devices and internet services get cheaper, with the evolution of technology. Therefore, it is getting easier and easier to develop new language skills, as it is getting easier to get in touch with foreign content through the web and mobile applications.

The language learning assisted by devices is a field of research divided mainly into the Computer Assisted Language Learning (CALL) and Mobile Assisted Language Learning (MALL).

With the ease to access the web, foreign language students can explore the contents in the language of their interest. There are many entertainment forms that can be converted into foreign language learning and practice, like music, movies, news or books. A vast amount of contents in the web is generated everyday, being text like news and blogs, or spoken language like in videos, web radios and podcast.

A part of the foreign language content in the web is associated somehow with the native language of the learner, such as lyrics translations and movie subtitles. Even if there is no association between the languages, the learner will be able to reach that association through translation tools, like the Google Translate[1].

However, even having the means to explore the language contents, a learner might feel lost in the wilderness of the web. Besides, languages are seen as hard to learn, as shown in the survey presented by newspaper The Guardian[2]. The young people interviewed

[1] Google's free online translation service for instantaneously translating texts and web pages, supporting more than 70 languages. Available on https://translate.google.com/

[2] Available on https://www.theguardian.com/education/2014/nov/07/-sp-do-young-people-care-about-learning-foreign-languages-data

in UK cite the difficulty of learning grammar as the main downside of learning foreign language and the help with grammar is the third point that would encourage them to learn a language. They also cite language apps, foreign films/TV and music as factors to encourage them to learn foreign language.

Computers could guide the learner through the contents, delivering contents according to his/her knowledge in the target language and to personal tastes. However, it is a challenging task for a computer application to drive the learner through the right contents, exploring the grammatical rules available in the contents and assuring that the learner is making progress through exercises, listening and speaking.

1.1 Objectives

The main objective of this research is to develop a framework for studying of verb tenses of the English language, through the exploration of English content by automatic exercises generation. It is part of a project that aims to help people to learn and develop knowledge in English language. It is designed to be used individually or with teacher orientation. CALL with teacher-driven instruction brings better results for student than only computer-based learning or only teacher-driven instruction (Kılıçkaya, 2015). Previous studies compared computer-based and teacher-driven acquisition of verb tenses, where the computer-based method slightly outperformed the teacher-driven approach, although both methods were effective (Rabab ah and AbuSeileek, 2009).

This research aims to provide a tool for the English language student to explore English content in order to learn and to practice the grammar of the verb tenses, in the future. For this purpose the tool must attend three minor goals. The first one is fundamental for the success of the project: to classify verb tenses precisely. Without precisely classifying the verb tenses it would be impossible to succeed on the next minor goals. The second minor goal is needed to assure and to measure the progress of the learner through the lessons: automatically build exercises based on a given content chosen by the learner. At last but not least, we need to rank the exercises generated from a text, so that exercises can be delivered ordered by difficulty.

1.2 Motivation

In the Mobile Assisted Language Learning (MALL), most of the studied topics, between 2000 and 2012, are focused in vocabulary, usability, perception/attitude and potential

uses/drawbacks (Duman et al., 2015). Considering that number may be similar to the Computer Assisted Language Learning (CALL), we can say that grammar needs more attention. However, it is a challenge to classify and organize all the grammar topics for delivering it to the learner. This work focuses on a part of the English grammar: the verb tenses.

The default contents in the formal second language courses are considering boring. The possibility of customization of contents, according to the themes that the learner considers interesting, may be a relevant factor for motivation and better performance. Another point is that the content will bring a vocabulary matching the learner interests.

1.3 Text Organization

This thesis brings five more chapters. Chapter 2 shows a review of readability estimation of texts in English for native and non-native speakers. Readability is an important measure for text selections for a second language learner and for ranking automatically generated exercises. The related work is presented in Chapter 3. In Chapter 4, the methods and the system architecture are presented. In Chapter 5, the experiments and results are described and analyzed. The conclusions and future work are presented in Chapter 6.

Chapter 2

Background

This chapter presents the concepts for readability estimation for native speakers, in Section 2.1 and readability estimation for ESL learners in Section 2.2. Readability is not the key of this research, however it is key for ranking the generated exercises to deliver to the leaner, always in a ascending difficulty. Delivering a hard exercise for a learner may be lead to frustration. In the other hand, the learner would get bored facing only too easy exercises (Beinborn et al. (2012))

The readability of a text has been researched for several decades. There are thousands of references and hundreds of formulas have been defined (Collins-Thompson, 2014). McLaughlin (1974) defined readability as the text quality that makes the reader interested to keep on reading. For Dale and Chall (1949), the readability is the sum of the elements in a text that will affect on comprehension, reading speed and the interest on the text. These elements are text features that may fit in one of the following categories, defined by Beinborn et al. (2012):

- Lexical: It is associated with the difficulty of a word. Rare and ambiguous words increases reading difficulty.
- Morphological: Features extracted by the analysis of the word composition, such as prefixes or suffixes.
- Syntactic: Features related to the sentences structures, like the number of verb phrases.
- Semantic: It is related on the meaning of words or verb phrases that can vary depending on the context that it is applied.

- Discourse: Features extracted from cohesion and coherence elements.
- Conceptual: When the text needs previous knowledge to be understood.
- Pragmatic: The interpretation of the text requires reading skills and/or background knowledge, as the cases of satire or irony.

2.1 Readability for native speakers

This section describes readability and several formulas and metrics that were defined to measure it. The readability formulas were based on metrics such as average sentence length, average number of syllables, frequency of the words contained on the sentence and length of words in chars.

Initially, readability metrics were based on frequency of use of the words and sentence and word length. The first 300 most frequent words make up about 65 percent of all written material (Wang, 2016), so that the more frequent the words of a text are, more readable the text is. A list of 10,000 most frequent words was made by Thorndike (1927) and it was used in further studies, like in the development of the first readability formula created by Lively and Pressey (1923) to provide a method for helping junior high school students to select science books at adequate levels. The evaluated metrics were the number of different words, the number of words not present in the Thorndike list and median index number of the words in the Thorndike list. They concluded that the best indicator of readability was the median index number.

The Reading Ease formula was developed by Flesch (1948), based on the average number of syllables per word and the average sentence length. It was found previously that sentence and word length are two important features for estimating readability (Kitson, 2010). The Flesch formula is shown in Formula 2.1. Being applied to reading tests, the multiple correlation coefficient value of the formula was 0.7.

$$GL = 207 - avgsentencelen - 85 \times avgnumsyl \tag{2.1}$$

Dale and Chall (1948) used the concept of familiarity of a word and its average sentence length for the definition of their readability formula. The non-familiar words are the ones that are not in the Dale-Chall vocabulary (*!freq3000*) that lists of the most frequent words of the English language. The formula is presented in Formula 2.2.

$$GL = 0.16 \times !freq3000 + 0.05 \times avgsentencelen + 3.6 \tag{2.2}$$

In order to analyze new metrics, Bormuth (1969) conducted a cloze test with junior students with texts from student's textbooks. 24 new readability formulas were created by Bormuth and it was found that new variables did not bring a great contribution on predicting readability, comparing with the sentence length and vocabulary features.

Kincaid et al. (1975) used the average sentence length with the average number of syllables to predict the difficulty of a text. On the other hand, Coleman and Liau (1975) defended that the length of the words in chars as a better metric to predict readability when compared to the number of syllables.

The percentage of words with 3 syllables or more was used as text difficulty predictor, with average sentence length by Gunning (1952). The average number of words with 3 syllables or more was also use by McLaughlin (1969), on the SMOG formula.

The formulas created for readability showed correlation with the text difficulty(Bormuth, 1969; Coleman and Liau, 1975; Dale and Chall, 1948; Flesch, 1948; Gunning, 1952; Kincaid et al., 1975). However, the metrics used are based on surface linguistics features, and even being able to predict text readability, they are unable to predict other important aspects that affects text comprehension, such as cohesion, discourse-based features, presentation and writing style. This way, a text with random frequent words, with short sentence length, could be graded as easy to read having no meaning at all, which was shown by Young (2011). Another point is that the classic readability measures are unreliable for text passages with less than 300 words (Kidwell et al., 2011) and do not correlate well with the readability of texts from the web and other documents with non-traditional text style (Si and Callan, 2001).

The limitation of the classic approaches and the opportunities to apply new computation methods led researchers to explore machine learning techniques for better prediction of readability and exploration of higher level text features (Collins-Thompson, 2014).

Si and Callan (2001) presented a method that utilizes statistical language model, so that the frequency of each term of the corpus is taken into account to estimate the readability. Traditional features, such as average sentence length, were combined with a unigram language model. The experiments showed that the average sentence length is a good measure for predicting text difficulty, but the average number of syllables is not. For the experiments, 91 web pages were used with their readability level, divided

into 3 groups. 10 pages of each group were used for training, the other 61 were used to test. Formula 2.5 combines the language model with the average sentence length for the text difficulty prediction. $P_a(g|d_i)$ represents the language model and is given by the Formula 2.3, where $P(w|g)$ is the probability of a word w, in the document d, being on a given readability grade g. $P_b(g|d_i)$ is the average sentence length model, showed in Formula 2.4, having as parameters $s1$, the length of each sentence, $s_{|g}$, the average sentence length on each readability level, and σ is the standard deviation of the length of sentences. λ is the weight parameter to balance the models. The best value of λ was 0.91, which shows that the language model is more significant. With this value of λ, the reached accuracy was 75.4%.

$$G = argmax \sum_{w \epsilon d} log(P(w|g)) \tag{2.3}$$

$$G = argmax \sum_{s \epsilon d} log \left(\frac{1}{\sqrt[2]{2\pi}\sigma} e^{\frac{-(s_1 - s_{|g})^2}{2\sigma^2}} \right) \tag{2.4}$$

$$P_c(g|d_i) = \lambda * P_a(g|d_i) + (1 - \lambda) * P_b(g|d_i) \tag{2.5}$$

Collins-Thompson and Callan (2005) used an unigram language model for the task of text difficulty classification which considers the capability of prediction of each word, based on its frequency. It was found that certain words has greater prediction capability for specific classes. The authors show that readability measures may be created using the statistical language model.

Schwarm and Ostendorf (2005) concluded that trigram statistical language models perform better than bigram and unigram models, for readability prediction. They combined the language model approach with support vector machine approach, providing the best results. It was also considered syntactic features, besides the language model.

Anagnostou and Weir (2006) explores the idea of using the frequency of collocations to measure text readability. A collocation is a sequence of two of more words with a unique meaning and features of a single semantic and syntactic unity. Some types of collocations in English are the following:

- Verb + noun, e.g. *commit suicide.*
- Adjective + noun, e.g. *reckless abandon.*
- Adverb + verb, e.g. *tacitly agree.*
- Verb + prepositional phrase, e.g. *set in motion*

The proposed measure of readability based on collocations is shown in Formula 2.6, where acf is the collocation average frequency, n_c is the total number of occurrences of collocations in the text in analysis, m is the number of different types of collocations in the text in analysis, f_i is the frequency of the collocation of the type i in *corpus* and n_i is the number of occurrences of the collocation i in the text in analysis.

$$acf = (1/n_c)(\sum_{i=1}^{m} f_i * n_i) \tag{2.6}$$

Callan and Eskenazi (2007) combine lexical and grammatical features (22 grammatical constructions) to improve readability prediction for native students and foreign students. The authors believe that the vocabulary based approach is better than the grammatical approach, but the both approaches combined bring better results than each of them apart. The grammatical structures are found by the search of patterns (rules). The kNN (k-Nearest Neighbor) algorithm was used for the classification based on grammatical features. The Euclidean distance was used to measure the distance between the features vectors. Table 2.1 shows the occurrence rate of some grammatical features. The result of the interpolated model fro readability prediction is given in Formula 2.7, where L_{LM} is the language model prediction, L_{GR} is the kNN prediction based on grammatical features, and C_{kNN} is the confidence value used for the kNN algorithm.

$$L_I = L_{LM} + C_{kNN} * L_{GR} \tag{2.7}$$

Heilman et al. (2008) compared several statistical models and features for readability classification. The best set of features could not be found, however, it was observed that all the features combined resulted in better correlation coefficient and better accuracy. The analyzed models were the Linear Regression, the Proportional Odds and Multi-class Logistic Regression. The Proportional Odds performed better on the experiments.

Table 2.1: Features occurrence rate per hundred of words used for text readability grammar-based prediction. Source: Callan & Eskenazi (2007).

Feature	Lowest Level	Highest Level
Passive Voice	0.11	0.71
Past Participle	0.28	1.63
Perfect Tense	0.01	0.33
Relative Clause	0.54	0.60
Continuous Tense	0.19	0.27
Modal	0.80	1.44

Feng et al. (2010) analyzed a variety of readability estimation features, grouping them in model language, part-of-speech (PoS), traditional, syntactic and discourse categories. The analysis shows the ability of each feature of each group to predict text difficulty and the ability of each category combining the features. It was observed that PoS categories highly correlated with text difficulty and the nouns play important in this correlation.

Even with several decades of research, some points need to be improved for readability research progression (Collins-Thompson, 2014):

(a) Improved annotated corpora: It is hard to find a corpus for readability research. The problem also occurs to find a readability corpus for ESL learner Xia et al. (2016). Building a corpus involves several parameters, as labelling the text difficulty levels by many people and copyright issues. A corpus correctly built would allow experiments to be replicated for methods validation and algorithms comparing. Base de dados anotada. É difícil encontrar um *corpus* disponível para pesquisa, principalmente por questão de direito autoral.

(b) Standard task definition and evaluation of methodology: It is necessary to standardize the task definitions and evaluation criterias, as it is done in other areas such as summarization and entity recognition.

(c) Inter-disciplinary collaborations: Individual text difficulty modeling and text comprehension are not a problem only for computer science, it also depends on other fields, such as linguitics, education and psycology

2.2 Readability for ESL learners

Readability is also researched in order to assess ESL (English as Second Language) students. Brown (1997) made a experiment to analyze the readability measures applied to ESL students. Fifty text passages were randomly selected to build cloze tests. Every 12th word was removed from the passage. The cloze tests were applied on 2,300 Japanese English learners. It was found that the indices used to predict readability for native learners does not predict readability the same way for ESL students. The new formula brought by Brown, was better associated with the results of the cloze experiments than the traditional readability metrics, which were focused on foreign speakers. It has reached a multiple correlation of 0.74 and 0.51 of R^2 with multiple regression.

In contradiction, five classic readability formulas had good performance on predicting text difficulty for Japanese English learners, in experiment realized by Greenfield (1999). The disagreement was given by the text passages used, as pointed out later by Greenfield (2004). Brown's study selected random passages from written books found in the U.S public library, in order to be representative, while Greengied tested the classic formulas on academic texts in order to compare the text difficulty for ESL learners and native English speakers.

For Beinborn et al. (2012), the readability measures, considering the self-directed learning, must be suitable to target the specific learner, instead of applying a generic readability formula. The learner background knowledge should be taken into account and a proper model be applied. To evaluate the learner background, two approaches would be considered. The first is the input-driven method which would be based on how well the learner would perform on exercises generated on texts that learner wanted or had to read. The second is the curriculum method where a learning goal is predefined, and then exercises are generated on the learning goal content.

Huang et al. (2011) used several lexical and grammatical features to train a linear regression model to predict the difficulty of texts for second language learners. For the experiments, two corpora were used. One was based on high school textbooks for Chinese students with six grades of difficulty. The second corpus had news web documents with their difficulty labeled by high school teachers. The reached results have outperformed previous methods and the estimation was close to human annotations.

Xia et al. (2016) collected a dataset from five main suite Cambridge English Exams[1] which are target in learner levels, and Weebit corpus Vajjala and Meurers (2012), which

[1] http://www.cambridgeenglish.org

is a large dataset, obtained from original authors, for readability analysis and contains texts targeted to several age groups. For the experiments, several dimensions of features were included:

- **Traditional Features**: Number of sentences per text, average and maximum number of words per sentence, average number of characters per word and average number of syllables per word. It was also included two popular readability formula, the Fesch-Kincaid score (Kincaid et al., 1975) and the Coleman-Liau formula (Coleman and Liau, 1975).
- **Lexico-semantic Features**: *English Vocabulary Profile* (EVP) is an online vocabulary resource with information on how learner acquire words and phrases. The lexico-semantic features were based on EVP.
- **Parse Tree Syntactic Features**: It includes average parse tree depth, average number of noun, verb, adjective, adverb, prepositional phrases and clauses per sentence and complexity measures based on grammatical relations.
- **Language Modeling Features**: Statistical language models based on n-gram of words and PoS tags.
- **Discourse-based Features**: Entity density, Lexical chain and Entity grid features. Entity density is strongly associated with text comprehension (Feng et al., 2010) and includes features like number of entities per document, the average number of entities and percentage of named entities. Lexical chain represents the semantic relations between entities in the text. Entity grid is an entity-based approach to measure text coherence presented by (Barzilay and Lapata, 2008). The grid is an array of two dimensions with information of the distribution of discourse entities across the text.

Xia et al. (2016) achieved the state-of-art in readability estimation, using Support vector machine (SVM), with a 0.803 accuracy and 0.900 Pearson's correlation coefficient, and a 0.785 accuracy and 0.924 Pearson's correlation coefficient. Future work on readability may consider the learner native language as a feature to predict readability to that specific user, as the language cognates will increase readability score (**?**).

Chapter 3

Related Work

The major area related to this project is language learning assessed by computer and mobile devices. Several ways to provide knowledge in foreign language through computers and mobile devices have been researched, and some of them are presented in this chapter.

3.1 Computer Assisted Language Learning and Language Tutoring Systems

The main goal of a foreign language learner is to reach or improve the ability to communicate in the studied language. Aiming that goal, the learner is trained to write, speak, read and listen, in autonomous way or with teacher. The evolution of a learner of a language can be measured, as the learner becomes able to solve more complex language exercises and to express the ideas and feelings, reaching the communication level. Automatic exercises generation allows the learner to study on his own pace, with dynamically selected content.

The types of automatic generated exercises presented here are cloze (fill-in-blank exercises) and automatic questions. In a cloze, parts of a text passage are removed and the student is prompted with the alternatives to fill the blanks. Not only the words to be removed must be analyzed, but also the wrong choices, called distractors.

Brown et al. (2005) described the generation of cloze for six types of questions, based on WordNet data, where the English words are grouped into their synonyms sets. They showed that "*the computer-generated questions give a measure of vocabulary skill*

for individual words that correlates well with human-written questions and standardized assessments of vocabulary skill".

Heilman and Smith (2010) presented an approach for question generation, based on a statement, for education assessment. The sentence parse tree is analyzed in order to decompose the main verb and invert the necessary words to form a question structure. As the example provided by the authors: *During the Gold Rush years in northern California, Los Angeles became known as the "Queen of the Cow Counties" for its role in supplying beef and other foodstuffs to hungry miners in the north.* A question that could be generated from the passage is *What did Los Angeles become known as the "Queen of the Cow Counties" for?* In the approach, candidates for question are rule-based generated. Of all generated questions in the experiments, 27% were considered acceptable. After applying a ranking model on the candidates, the acceptable percentage jumped to 52% when selecting the ranked 20% of questions. The ranking model was trained based on set of features grouped in the categories: length, words, negation, n-gram language model, grammatical, transformations, vagueness and histograms.

Heilman et al. (2006) described the REAP system and its challenges as an intelligent tutoring system. It gathers documents from the web and selects one for lesson, which will bring new information to the user. The documents selection are based on several features, as user vocabulary list, sentence length, target vocabulary list and document length. The classification of the documents takes into account syntactic features, readability, length and occurrence of the target vocabulary. After the user finished reading the text, a cloze exercise, automatically generated, will be applied on the user. This kind of system allows the user to learn on its own pace. The REAP software was used in classroom during a semester, when the users showed progress on completing the cloze tests. The REAP software does not have a focus in any part of grammar, unlike the system we present.

As challenge to that kind of automatic tutoring systems, Heilman et al. (2006) cite the difficulty to have good texts to learner. First, the documents are filtered, in order to keep texts with correct sentences and select texts which will bring novelty. In the experiment, after filtering the documents, only 0.5% of all documents remained. Another difficulty for automatic text selection is choosing the right texts without making it boring or offensive for the students, as each student has his/her own culture and traditions.

The computer generated clozes can test and improve the learner vocabulary in a language. It does not make the him/her to memorize a translation or synonym, but learn the word related to the context it is inserted. The clozes can explore several points

of the language, as prepositions or verbs. The automatically generated questions are also important for the student learner to think and to structure an answer. However, it is still a hard task for a computer to automatically correct the written answer of the learner. For instance, the shared task of the Conference on Natural Language Learning (CoNLL) 2014, was grammatical error correction and detection. The winner team had the F-measure of 37.33%.

This work is similar to the ones presented here because of the focus on the automated language learning and on tutoring systems. However, it is directed for the verb tenses acquisition.

3.2 Verb Tenses

Most of the verb tense works are focused on machine translation and on automatic error correction. In machine translation, there is a big challenge to keep the right meaning on translations of verb tenses regarding that each language has its own rules for verb tenses. This way, a verb tense in a language may not have correspondent in another language. The lack of data to map verb tenses from one language to another is also a problem (Lee, 2011).

Loaiciga et al. (2014) presented a method for verb phrase alignment between French and English to improve statistical machine translation (SMT), based on PoS tags, parser and heuristics.

In (Lee, 2011), it was developed a method to classify verb tenses and time expressions based on syntactic and lexical features. A statistical model was trained to be used in SMT. The work could not be used as a baseline as it gave no detailed results for the verb tenses classification.

Automatic error correction tasks can provide instantaneous feedback for a language learner when he/she is writing an essay. In (Tajiri et al., 2012), all the context is taken into account to the task of automatically correcting verb tenses errors. It was found that the most common mistake is using present simple instead of past simple. Verb tenses information is also used in the task of classifying temporal relationship and ordering of events in extraction of texts (Bögel et al., 2014; Derczynski and Gaizauskas, 2015).

Chapter 4

Methods for generating exercises

This chapter presents the proposed methods for identifying verb tenses in English texts and for generating verb tenses exercises. Figure 4.1 presents an overview of the components of the system. The input of the system is a text. Once the text is submitted to the system, it processed by the PoS Tags Extractor, having each verb tense classified into one of the 13 implemented verb tenses. With the classified verb tenses, the sentences are ready to feed the Exercises Generator. There are four types of exercises being generated. One of them uses a verb tenses index, which is built from external data. The data to be studied by the learner will follow the pipeline of components.

The Contraction Expander component (1) is responsible for expanding the unambiguous language contractions. Contractions are expanded because they lower the PoS tags classification precision. For instance, because of the tokenization, the word "won't", may be misclassified as the past form of verb "win", whereas the word "haven't" may be classified as a noun. Some words result in the same contraction, causing them to be ambiguous, like "had" and "would", whose contraction is "'d".

After the text is expanded, it will be provided as input to the PoS Tags Extractor (2), which is a wrapper for the StanfordNLP (Manning et al., 2014), the chosen NLP tool for the experiments. Here, the text is transformed into arrays of PoS tags, lemmas and words. Then, for each sentence, an object is created with the arrays of words, PoS tags and lemmas in the sentence scope. We refer to that object as a sentence instance.

Having the sentence instance, the Verb Tenses Pipeline (3) processes the periods, extracting their verbs, which will be processed by the Person Detector. The Person Detector (4) identifies the grammatical persons present in each sentence instance. Thus, the sentence instance is ready to generate exercises, by the Exercises Generator (5), which is composed by four specific types of exercise generators.

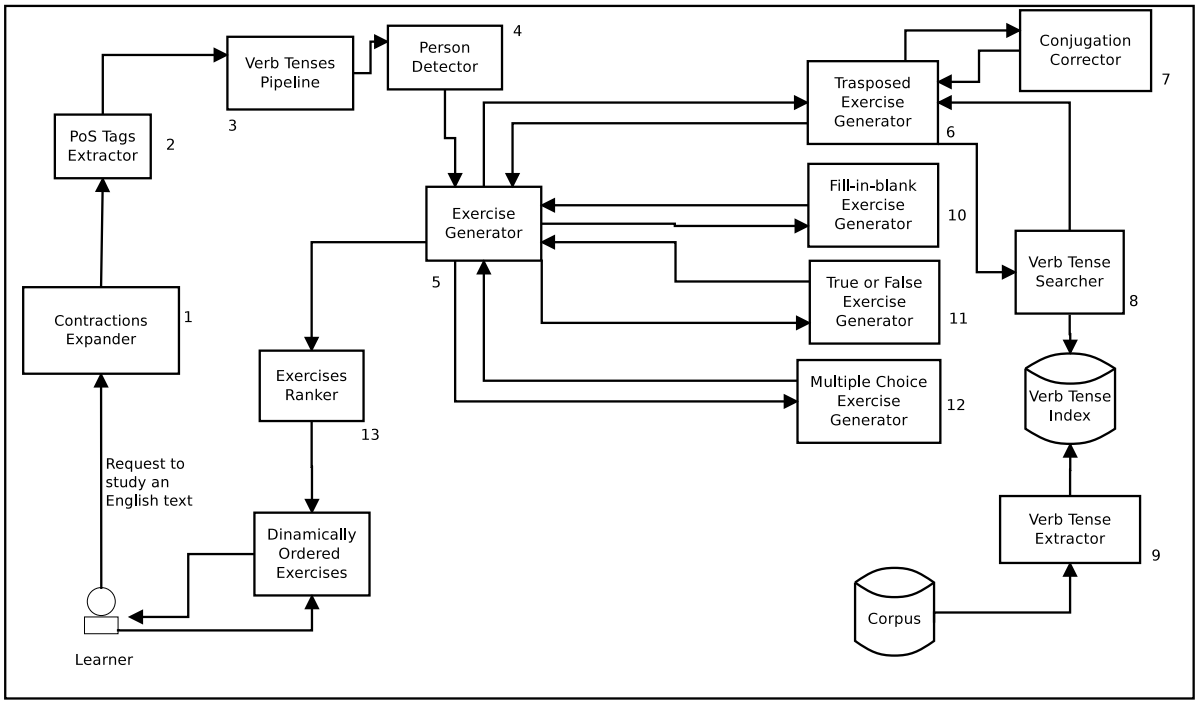

Figure 4.1: Top level view of the system architecture

The first exercise type is the Transposed Exercise, generated by the Transposed Exercise Generator (6). This type of exercise has two variations. In the first variation, the student must transpose the original sentence to a requested verb tense so that the system shall check wheter the transposed sentence matches the request. In the second variation, the student must transpose the automatically transposed sentence to the original verb tense (the text that appeared in the original text). The Transposed Exercise Generator relies on the Conjugation Corrector (7) and on the Verb Tense Searcher (8).

The Conjugation Corrector is the component responsible for checking and correcting the transposed verb tenses. It makes the verbal agreement correction. The verb tense is transposed by searching a target verb tense structure in the Verb Tense Searcher and, then, adapting it to the context. The Verb Tense Searcher uses the Verb Tense Index, built by the Verb Tense Extractor (9). The Verb Tense Index contains pieces of sentences associated with its verb tenses. Brought in search results, the verb tenses, with the respective PoS tags, are used as example of the structure for the target verb tense transposition.

The fill-in-blank exercise, built on component Fill-in-blank Exercise Generator (10), is a kind of exercise where part of the sentence is removed, and the options are given to the student to fill the right answer in the blank left. The options may be displayed

either in parenthesis, after the blank mark, or in a list with all options before all the sentences containing the blanks.

The True or False Exercise Generator (11) builds the type of exercise where, given a list of sentences, the student must assign, for each sentence, true, when the sentence is correctly written, or false, when the sentence contains any error. On the other hand, in the multiple choice exercises, the student must analyze the given alternatives, and choose the right or wrong one, as requested. This type of exercise is built in the Multiple Choice Exercise Generator (12).

With the exercises ready, the Exercises Ranker (13) is able to rank, by degree of difficulty, all the generated exercises. The student may be a beginner at a specific grammatical rule, and expert at others. As students use the system, completing the tasks, the exercises may be dynamically rearranged according to their hits or misses.

The details of each component presented in Figure 4.1 are given in the following sections.

4.1 Contractions Expander

The Contractions Expander (1) is the first component in the system pipeline. This component maps and transforms the contractions of the English language into their expanded form. Contractions are expanded to prevent PoS tags errors in the classification of PoS tags. Each contraction is mapped to its expanded version and is treated as a regular expression to be replaced on the text. However, there are three ambiguous text contractions, which are "'d" being contraction for "would" or "had", "'ll" being contraction for will and shall, and "'s" being contraction for "is" and "has". So far, only the unambiguous contractions are expanded, because it is relying only on a simple contractions dictionary, without any extra rules. The context of words and PoS tags are features to be explored, in order to solve the ambiguous cases.

4.2 PoS Tags Extractor

The PoS Tags Extractor (2) encapsulates the tool StanfordNLP. PoS tag is short for part-of-speech tag and indicates the grammatical class of a word. The PoS tags are used in the definition of the rules for the classification of the verb tenses.

4.3 Verb Tenses Pipeline

The Verb Tenses Pipeline (3) is the component responsible for identifying the verb tenses in a given text. It is composed by rule-based subcomponents. Each subcomponent represents a grammar rule which in our case means a rule for verb tense classification. The text received by the Verb Tenses Pipeline, just processed by the PoS tags Extractor, contains PoS tags and lemmas associated with the respective words. It is split into sentences, and the subcomponents test each sentence to classify it into one of the 13 verb tenses, present in the English Language.

The subcomponents in the Verb Tenses Pipeline are named according to the verb tenses names. So the subcomponents are: PresentSimple, PresentContinuous, PresentPerfectSimple, PresentPerfectContinuous, PastSimple, PastContinuous, PastPerfectSimple, PastPerfectContinuous, FutureSimple, FutureContinuous, FuturePerfectSimple, FuturePerfectContinuous and FutureGoingTo. The GrammarRule is the generic component for the verb tenses and contains the generic attributes and methods. All the subcomponents are an extension of the GrammarRule object and override the 'isRule' method.

The 'isRule' method checks if a given slice of the sentence is a valid match for the subcomponent. The Verb Tenses Pipeline component contains an array of the type GrammarRule. Each GrammarRule object in the array represents a single verb tense. When a sentence is given to the Verb Tenses Pipeline, it will be iterated over, and for each index of the sentence iteration, the GrammarRule array will also be iterated and the sentence will be tagged as a verb tense when the GrammarRule returns true on the 'isRule' method. If the rule of the verb tense is matched, then the part of the sentence is tagged with the verb tense.

The order of the GrammarRule objects in the array is fundamental for the classification success because the longest rules may contain smaller rules. For instance, the future simple expression "will have" would be classified as Present Simple if the Present Simple rule was matched before checking the Future Simple rule because it contains the verb "have" on its present conjugation, which is indicated by the PoS tag "VB" or "VBP".

Each GrammarRule in the array has its own max window size attribute, which sets the max number of words accepted in a rule. If a big window size is set to a rule the should have a small window size, it might include other verb tenses to analyze and cause classification errors.

The Verb Tenses Pipeline algorithm is showed in Algorithm 4.1. The array verbTenses contains the verb tenses objects, ordered from the longest rules to the smallest ones.

Then, the sentence being analyzed, is iterated over backward, and if a verb tense is matched, then the iteration index skips to the index just before the beginning of the matched rule. On each iteration, a PoS tag is instantiated with the current index. If it is a verb, then the verb tenses array is looped, and each rule object checks on which verb tense that part of the sentence will fall. At the end of the algorithm, the sentence will have its verb tenses classified.

Algorithm 4.1: Verb Tenses Classification in the pipeline

```
function getVerbTenses (sentence);
Input: A sentence instance with the respective PoS tags and lemmas
Output: A sentence with the respective verb tenses and their indexes of start
  and end
verbTenses = [FuturePerfectContinuous, PresentPerfectContinuous,
  PastPerfectContinuous, FutureGoingTo, FutureContinuous, PastContinuous,
  PresentContinuous, FuturePerfectSimple, FutureSimple, PastPerfectSimple,
  PresentPerfectSimple, PastSimple, PresentSimple]
for i = sentence.lenth; i >= 0; i -- do
    postag = sentence.postags[i];
    if postag is verb then
        for vTense in verbTenses do
            if vTense.isRule(sentence, i) then
                sentence.tag(vTense, vTense.startOfRule, vTense.endOfRule);
```

4.4 Person Detector

Component Person Detector (4) detects the grammatical person for each verb tense. It analyzes each verb tense and its preceding words to determine the grammatical person. On the verb tenses transposition, it is necessary to know the grammatical person, so the transposed verb tense can be properly conjugated to agree, verbally, with the subject.

The Person Detector is also a rule-based component and it matches grammatical persons using regular expressions. Classifying grammatical persons is not such a trivial problem when we have compound subjects, articulated with 'and', 'or' and possessive pronouns. All the cases were treated using rules that analyzes words, PoS tags and lemmas. For instance, if the PoS Tag of main verb of the verb tense in analysis is "VBZ" or the auxiliary verb is "does" or "has", we can say it is third person.

4.5 Exercises Generator

As our system aims to provide an interface for users to explore English texts for learning and fixing the language concepts, we also worked on the automatic exercises generation. Exercises are the way to measure the learners level, and guide them through the concepts they need to know. Those are identified by the way users complete exercises, either by hit or mistake. Component Exercises Generator (5)is the top component for the tasks of generation of exercises.

For each category of exercises, there are different levels of difficulty. Therefore, the user can be exposed to harder or easier exercises. The levels of difficulty are given based on the frequency of the verb tense in the corpora and on the frequency of the verb tense variation. The core of a verb tense may vary structurally, as it may contain other grammatical classes besides verbs, such as adverbs and pronouns. We consider the most frequent verb tenses and verb tenses variations being the easier levels.

Once the user finishes an exercise, the answer is processed by the system. If the user makes mistakes, the next exercises are recalculated, in order to select the ones that match the user level, and the option to see more about the grammatical concept for that exercise will be prompted.

We propose four types of exercises, which are described in the following sections. They are about verb tenses transposition, fill-in-blank, true or false and multiple choice exercises.

4.5.1 Transposed Exercise Generator

The transposition of verb tenses is an exercise to force the learner to think in a different verb tense and build right sentence structures for it. This kind exercise is generated on the Transposed Exercise Generator (6) and can be provided to the user by two different methods:

1. presenting an automatically transposed sentence, asking the user to transpose a given verb tense to its original verb tense, as it appeared in the original text.
2. asking the user to transpose the original verb tense directly to another.

In the first method, a sentence is automatically transposed to random verb tense, if it is possible to transpose to that target verb tense. The transposition starts with the

selection of the part of the sentence to be transposed and its target verb tense, which must be different of the current verb tense of that part. The main verb is extracted from the part to be transposed and its lemma is used with the target verb tense as parameters to search, through the component Verb Tense Core Searcher, the transpositions candidates. If there are no matches for the search, the Verb Tense Core Searcher will try to find a candidate using the PoS tags structure of the text to be transposed, besides the target verb tense. If no results are found, then it is considered impossible to transpose that part of the sentence.

The transposition process takes as parameters a sentence instance, a verb tense in that sentence, and a target verb tense. The sentence instance has, as attributes, the NLP features and the previously classified verb tenses. The first verb tense parameter will be referred as the original verb tense, and the second as the target verb tense. The main verb is extracted from the original verb tense, to be used as parameter to search. Let us consider the sentence "I have always been working here", whose main verb is "to work", as an example for the transposition process.

The sentence is in the Present Perfect Continuous form and its original verb tense for our application is considered to be "have always been working". A search will be done taking as parameters the targeted verb tense, the non-verb PoS tags in the original verb tense and the main verb. The only non-verb PoS tag for our case is the adverb "always", the main verb lemma is "work" and the target verb tense is Past Simple.

The results of the search will be candidates to the transposition. In order to validate the candidates, the algorithm will check if a candidate has the same non-verb PoS tags and then inject the original non-verb words in the respective places, mapped by the PoS tags. For instance, "normally worked" would be a valid candidate for our transposition, because an adverb can be replaced by another adverb, so that adverb "normally" would be replaced by "always" and the result would be "I always worked here". At last, the algorithm corrects conjugation errors. In transposition to Present Simple, a conjugation error would be "I always works here".

In the case when no results are found, a new search is made, ignoring the main verb. So, we would search only for Past Simple and for non-verb PoS tags. To be a valid candidate for the transposition, the correct form of the main verb must also be searched and found, thus, the candidate also needs to have the same non-verb PoS tags of the original verb tense. In this case, the candidate "normally walked" would be accepted and the verb "work" would be searched in the same form of the verb "walked" in the candidate. If found, the verb and adverb are replaced in the candidate, forming the final verb tense "always worked", which will replace the original verb tense in the sentence.

In the second method, the user is asked to transpose a specific part of the sentence to a specific verb tense. The user submits the transposed sentence and the software checks if the sentence was correctly transposed, analyzing if the transposed verb tenses matches the requested target verb tenses. Besides the part transposed, the rest of sentence must remain the same to be a valid transposition.

Verb tenses transposition, crudely done, may build nonsense sentences. We think it is better showing a nonsense sentence, asking the user to transpose it to make it right, than showing a correct sentence, asking the user to transpose it to a nonsense sentence. So we only provide the first method of generate transposition exercises. Let us consider the sentence "I am going to play soccer", which appeared in text read by the user. In the generation of a transposed verb tense exercise, the sentence "I have been playing soccer" is displayed to the user and he/she must transpose it to the Present Continuous, which is the original verb tense. When the user inputs the transposition, the software will check whether is matches the original one.

Verb Tense Core Extractor and Verb Tense Core Searcher

The Verb Tense Core Extractor (9) iterates over the corpora, extracting and indexing the identified verb tenses. The index is used by the Verb Tense Core Searcher, specifically in the verb tense transposition process. The indexed fields are the listed below.

- Verb Tense: the verb tense of the part of sentence to be indexed.
- Text: the original text.
- Lemmas: the original lemmatized words.
- PoS tags: the original PoS tags.
- Main verb lemma: the verb on its lemmatized form.
- Non verbs words: only the words that are not verbs.

When the documents are on the index, we will have data to make the verb tenses transpositions.

The Verb Tense Core Searcher is the search low-level component and it is used to find candidates for the verb tense transposition. It makes all the possible searches with the available parameters. At first, the search tries to find a candidate directly, specifying

the the verb tense, the main verb lemma and the PoS tags. If results are found and there are other type of words than verbs, such as pronouns or adverbs, these words are replaced with the respective ones, from the original text.

If there are not results, another search is made, but now, the main verb parameter is not used. So the search is made querying the target verb tense, the original text and the PoS tags, that means we are searching for a verb tense with same PoS tags structure but the main verb may be ignored.

If results are found, we have to analyze the main verb PoS tag, on the candidate. That PoS tag will be used to query the verb we want. So now, the third search has the goal to find the verb we want, conjugated in the target verb tense form. If results are found for the verb, and there are other grammatical type of words than verbs, these words are replaced with the respective ones, from the original text. The verb tense transposition is ready.

Conjugation Corrector

The component Conjugation Corrector (7) analyzes the transposed verb tenses in order to detect and correct the conjugation errors, as the search only brings a raw transposition result.

There is a pipeline in the component where on each stage, the previously classified grammatical personal and the kind of verb tenses construction are checked. The verb tenses construction may rely or not on auxiliary verbs. The auxiliary verbs are the verbs 'be', 'do', 'have', 'will'. So the pipeline process checks on which auxiliary verb the analyzed verb tense will fit, and then, which person will match. If a change is needed, the replace of the old wrong text is done.

4.5.2 Fill-in-blank Exercise Generator

The Fill-in-blank Exercise Generator (10) removes the verbs on the verb tenses in a sentence and shows the lemmas of the removed verbs with the original verb tenses as options for the user to fill the blank with the correct verb conjugation on the target verb tense.

The component builds two kinds of fill-in-blank exercise. In the first type, the verb lemmas and verb tenses are shown right after each blank, in parenthesis and split by

a slash. In the second type, a list with the options is firstly presented, then come the sentences with its blanks. The user must fill the blanks with the options on the list with right conjugation. There might be options fitting in more than one blank, without changing the meaning of the original sentence, but the user should know the right matches as he has read the text containing the sentences of the exercise. After the user submits the answer, the system will compare the submitted content with the original text.

4.5.3 True or False Exercise Generator

True or False Exercise Generator (11) is the component that builds exercises where the sentences must be assign with true or false. There are two kinds of affirmation prompted on this practice. In the first kind, a random grammatical person is assigned to a verb tense, the Conjugation Corrector checks that verb tense and alters the sentence, so there is a probability that the new grammatical person does not match the sentence, thus there is a chance to make a mistake in the sentence. Then the user must assign true, if the sentence is correct, and false if it is not.

In the second type, the verb tenses of sentences are listed and displayed with the corresponding sentence. The user must verify if the list matches the verb tenses of the studied sentence, in the order they appear. There is a chance of the system changing a verb tense on the list, making it unlike the original.

4.5.4 Multiple Choice Exercise Generator

The multiple choice exercise, generated by the Multiple Choice Exercise Generator (12), displays lists of verb tenses for each choice and only one choice is the correct list of verb tenses of the analyzed sentence. For generating the list with an incorrect verb tense, one verb tense of the list is chosen and substituted with a similar verb tense, by prefix or suffix, i.e. present continuous replaced with past continuous or present perfect simple replaced with present perfect continuous.

4.6 Exercise Ranker

The exercises are ranked based on language model, as it is the isolated feature that best correlates and with best accuracy for readability prediction (Xia et al., 2016). The

Berkeleylm tool[1] was used to train the language model with the data of the subtitles corpora.

[1]https://github.com/adampauls/berkeleylm

Chapter 5

Experiments

This chapter describes the experiments used to validate and to improve the process of identifying verb tenses and generating exercises about them. Section 5.1 presents results on the iterative process of building, improving and checking the verb tenses identification rules. The verb tenses transposition process was also iterative and its results are presented in Section 5.2. In Section 5.3 is shown how the readability may help to find misclassified verb tenses.

The experiments were done using corpus named Opus (Tiedemann, 2012), which is a corpus compound by several corpora from the web. We used specifically the OpenSubtitles corpus 2012 and 2013. It is set with more than 45 thousand documents and each document represents a movie, in which each document contains information about the time when each sentence occurs in the movie, besides the text. We believe that type of corpus can bring a big variety of samples, including formal and informal language.

5.1 Verb Tenses Detection

To start our verb tenses detection experiment, an initial set of rules were created for the 13 analyzed verb tenses. The verb tenses analyzed were Present Simple, Present Continuous, Present Perfect Simple, Present Perfect Continuous, Past Simple, Past Continuous, Past Perfect Simple, Past Perfect Continuous, Future Simple, Future Continuous, Future Perfect Simple, Future Perfect Continuous and Future with "Going To". It was generated a sample with 200 random sentences from random docs of the corpus to verify how the verb tenses classifier would behave. We could not assure neither those sentences would have at least one verb tense nor would them gather samples for all the verb tenses. It would depend on the verb tenses distribution in the corpus.

The first sample brought data mostly concentrated in the present and past simple verb tenses, with 145 samples for Present Simple and 39 for Past Simple. The next tenses with more samples were Present Perfect Simple, Present Continuous and Future Simple with 10, 9 and 9, respectively. Four verb tenses (Present Perfect Continuous, Past Continuous, Past Perfect Simple and Future Going To) only gathered 1 sample. The other verb tenses had no sample: Past Perfect Continuous, Future Continuous, Future Perfect Simple and Future Perfect Continuous. The data concentration is shown along with the precision and recall metrics in Table 5.1.

Table 5.1: First sampling for the verb tenses detection experiment

Verb Tense	Total	Precision	Recall
Present Simple	145	0.986	0.905
Present Continuous	9	1.000	0.111
Present Perfect Simple	10	0.700	1.000
Present Perfect Continuous	1	1.000	1.000
Past Simple	39	1.000	0.867
Past Continuous	1	1.000	1.000
Past Perfect Simple	1	0.000	0.000
Past Perfect Continuous	0	-	-
Future Simple	9	1.000	1.000
Future Continuous	0	-	-
Future Perfect Simple	1	1.000	1.000
Future Going To	0	-	-
Future Perfect Continuous	0	-	-

There were 185 verb tenses correctly classified. 55 were correctly classified as having no verb tense. 25 verb tenses were not identified: the misses occurred for classes Past Simple, Present Simple and Present Continuous. Three PoS tags errors led to verb tense misses. 9 verb tenses were wrongly classified. That makes the accuracy measure 87.59%.

Three out of the nine false positives should have been classified as Present Simple, but were classified as Present Perfect Simple. The errors occurred because the rule did not assume the cases of the verb be on the third person followed by a verb on the participle form, like in the sentence “It’s arranged by genre”. Thus the contracted form of “is” was considered as contraction of “has”. The rule was adjusted to consider also the verb lemma to match the Present Perfect Simple. One case was the expression ‘have got’

being classified as Present Perfect Simple when it should be put in the Present Simple group.

The cases of false positives and false negatives were treated, through the rules correction, and they were reduced from 9 to 5 cases for false positives and from 25 to 5 cases for false negatives, after rerunning the classification process for the same sample.

Over the iterations, there had been three types of errors:

- Classification error: There is a problem with the rule used to classify a verb tense, and it need to be fixed. It is the only type of error treated over iterations.
- PoS tag error: The used PoS tagger has classified a important word wrongly.
- Bad sentences constructions: The sample sentence has a wrong structure.

5.1.1 Verb Tenses Iterations 2-5

As we could not find enough samples for all the verb tenses by random selection, we decided to generate a fixed-number sample for each verb tense. Thus, instead of generating a large set with random sentences, we started to generate subsets of random sentences, being each subset a verb tense. Our goal was to fill each subset with 30 instances of verb tenses, totaling 390 verb tenses, when all the verb tenses are included. However, the sample may be smaller, when there are repeated sentences in the subsets or there are not enough samples for a verb tense. A verb tense is not included in a sample, once it has reached a precision higher than 90% in the previous iteration, because the rule would be considered correct. But it will be included in the sample again whenever its rule is modified by replication of changes from a similar verb tense. For instance, a rule needed to be added to the Present Continuous, then the rule could fit and be necessary for Past Continuous.

The first iteration has tested our rules and showed which one should be improved. Only the Present Simple and the Past Simple verb tenses would not need more iterations, since their precisions matched our goal, and they gathered enough samples. Iterations 2 to 5 were performed to sample, test and improve the other 11 verb tenses. The results are shown on Table 5.2.

On iteration 2, several types of errors were identified and corrected. The first rules allowed the verb tenses to be classified even when there was punctuation between the

extremes of the verb tenses. The rules were adjusted for this case. Another error was the combination of the verb on infinitive followed by gerund being classified as Present Continuous. This error occurred in sentences like “You’ve got to be kidding me”, and the Present Continuous would be marked in the words “be kidding”.

Still on iteration 2, nouns, pronouns, and determiners in the middle of verb tenses, for instance, would be accepted in the verb tenses rules for affirmative sentences. The following pieces of sentences would be classified as Present Perfect Simple before the corrections of the rules: “if you have any love left...” and “Maria had a very grown up...”, where the main verbs of the tenses would be “left” and “grown”, but they are not really part of the tenses, since the main verbs are “had” and “have”, respectively.

Infinitive also was accepted in the middle of the Past Perfect Simple, like in the sentence “If somebody had to get hurt...”. This sentence was classified as Past Perfect Simple because of the verb “hurt” in the participle form and the word “had” was found in the sentence. The rules were corrected for not accepting verbs between the main verb and the auxiliary verb “have” in the perfect tenses.

There were two similar cases of Future Simple being classified as Future Perfect Simple. The first one can happen in cases like the sentence “I will have it fixed”. The word “fixed” would be classified as the main verb for the Future Perfect Simple. This mistake can be avoided by checking the nouns, pronouns and grammatically similar words after the verb “have” and before the verb that would be the main verb, in this case, the verb “fixed”. The second case happens in cases like the sentence “you'll see that I’ve taken care” where the modal “will” was considered to be in the same verb cluster of “'ve taken” and the main verb was the verb “see”. This case was corrected by checking the existence of verbs between the modal “will” and the verb “have”. Those kinds of errors also happened on iteration 3, and the Future Perfect Simple rule was adjusted. The hits rate dropped for the Future Perfect Simple on Iteration 3 and it happened just because the random sample brought more the misclassified sentences.

Iterations 2 and 3 did not sample sentences for the Future Perfect Continuous because there was a maximum number of tries to get random sentences for each verb tense. Then, our strategy was changed to check all sentences, by iterating every sentence of all documents of the corpus, when the trials to get random sentences had run out. This way, iteration 4 brought the first sample of the Future Perfect Continuous. The sample gathered only 23 sentences with 26.08% of precision which is a very low result. The kind of error listed above for the Future Perfect Simple also occurred for the Future Perfect Continuous and Present Perfect Simple and were corrected in a similar way, by

Table 5.2: Rules Improvements - Iterations 2-5

Verb Tense	Iteration 2		Iteration 3		Iteration 4		Iteration 5	
	Prec.	Abs.	Prec.	Abs.	Prec.	Abs.	Prec.	Abs.
Present Continuous	0.900	30	-	-	-	-	-	
Present Perfect Simple	0.933	30	1.000	30	1.000	30	-	-
Present Perfect Continuous	0.967	30	-	-	-	-	-	
Past Continuous	1.000	29	-	-	-	-	-	-
Past Perfect Simple	0.900	30	0.933	30	-	-	-	-
Past Perfect Continuous	0.965	29	-	-	-	-	-	-
Future Simple	1.000	29	1.000	30	-	-	-	-
Future Continuous	1.000	30	-	-	-	-	-	-
Future Going To	1.000	30	-	-	-	-	-	-
Future Perfect Simple	0.700	30	0.500	26	0.965	29	-	-
Future Perfect Continuous	-	-	-	-	0.261	23	0.928	14

the application of the same rule. Iteration 5 was run to assure the goal for the Future Perfect Continuous was reached. The Future Perfect Continuous had 92.85% of precision with 14 sentences sampled. The samples show that the Future Perfect Continuous is rarely used, as we could not gather with 30 samples in any iteration.

5.1.2 Iterations for analyzing verb tenses transposition

Iterations 6 to 8 are relative to corrections done between iterations of the verb tenses transposition. The transposition experiments, which are presented in Section 5.2, showed several points where the rules needed to be improved in order not to bring errors from the classification process to the transposition process. Every change made on the verb tenses classification rules on a transposition iteration implied on the need to generate a new index for the Transposed Exercise Generator, as the transpositions are search-based and an index with wrong tenses would result in wrong transpositions. The results of iterations 6 to 8 are presented on Table 5.3.

On iteration 6 all the continuous tenses were modified for not accepting adjectives before the main verb on gerund. For instance, the sentence “It will be great talking to you” would be classified as Future Continuous before the rule adjustment because

"talking" would be considered the main verb. The same case would occur in the sentence "You need to be taking notes" being classified as Present Continuous. The Infinitive was separated from the Present Simple, as the transpositions would have to be adjusted specifically for the Infinitive.

All the perfect tenses were altered for not accepting verb on infinitive in the middle of the verb tense. Those cases occurred in sentences like 'he had to have been standing...' being classified as Past Perfect Continuous.

On iteration 7, respective to iteration 2 of experiments of the verb tenses transposition, more errors of classification of verb tenses appeared. The sentence 'It was someone asking for directions' would be classified as Past Continuous having the 'asking' as the main verb. The continuous rules were fixed and started to analyze the presence of nouns before and in the verb tense. The sequence of words 'had was' would be tagged as Past Perfect Simple, in sentences like 'All I had was...'. The Rule was adjusted for not validating the case as Past Perfect Simple. Errors due to PoS tags classification errors occurred for Present Simple and Past Simple.

On iteration 8, the use of gerund as infinitive was caught and treated. Periods like 'my earlier comments were just me having fun...' would be classified as Past Continuous. The solution was to check if the main verb, in the gerund form, is preceded by a possessive noun. Another words and grammatical classes also indicate that the gerund verb is being used as an infinitive, such as 'of' and 'for' preceding the gerund. The rule was added to the simple form of continuous tenses.

The final precision of each verb tense is shown in Table 5.4, where most of the verb tenses have reached more than 95%.

5.2 Verb Tenses Transposition

The verb tense transposition experiment was also made in an iterative way, so that the mistakes could be caught and the consistency of the corrections would be tested on the next iteration.

The samples generated for this experiment are given by the transposition of five random sentences of random documents for each verb tense to the other 12 verb tenses. Then, the samples would be compound of 780 transpositions. However, it is not always

Table 5.3: Rules Improvements - Iterations 6-8

Verb Tense	Iteration 6		Iteration 7		Iteration 8	
	Prec.	Abs.	Prec.	Abs.	Prec.	Abs.
Present Simple	-	-	0.967	30	-	-
Present Continuous	0.964	28	0.931	29	1.000	30
Present Perfect Simple	1.000	29	-	-	-	-
Present Perfect Continuous	1.000	29	1.000	30	-	-
Past Simple	-	-	0.933	30	-	-
Past Continuous	1.000	30	1.000	30	0.933	30
Past Perfect Simple	-	-	1.000	28	-	-
Past Perfect Continuous	0.952	21	0.965	29	-	-
Future Continuous	1.000	30	1.000	30	1.000	30
Future Perfect Continuous	0.875	8	1.000	7	-	-

Table 5.4: Last precision value for each verb tense

Verb Tense	**Precision**
Present Simple	96.7%
Present Continuous	100.0%
Present Perfect Simple	100.0%
Present Perfect Continuous	100.0%
Past Simple	93.3%
Past Continuous	93.3%
Past Perfect Simple	100.0%
Past Perfect Continuous	96.5%
Future Simple	100.0%
Future Continuous	100.0%
Future Perfect Simple	96.5%
Future Going To	100.0%
Future Perfect Continuous	100.0%
Average Precision	98.2%

possible to transpose from a verb tense to another specific verb tense and there might be cases where fewer samples are generated than the sample size goal. In the first iteration, 650 instances were correctly transposed and there were errors in 98 instances, totaling

86.9% of hits. The mistakes found in all transposition iterations are divided into the types:

- Agreement error: occurs when there are conjugation errors. Example of this error: 'He is going down' transposed to 'He go down'.
- Classification error: occurs when there are errors in the verb tense classification process. The wrongly classified pieces of texts goes to the index to be used in the transposition.
- Structure error: is a kind of classification error, and brings the right words in a wrong order on the transposition.
- Transposition error: occurs when there are errors in the process of transposing the sentence.
- PoS tags error: errors of classification of PoS tags implies in errors in verb tenses classification and transposition.
- Modal error: occurs when the transposition is done in verb tense which is related to a modal verb. The modal words were not considered for the experiments here. So transpositions of verb tenses related to a modal verb would be ignored.
- Missing auxiliary verb: in some transpositions, the verb 'do', and its past form 'did' were missing in negative forms.
- Main verb error: occurs when the main verb of the verb tense is missed or changed in transposition.
- Incomplete sentence error: occurs when the verb tense has an auxiliary verb but no main verb and the transposition is called.
- Been error: occurs when the been is inserted without need on transposition to Past Perfect Simple.
- Specific cases: the specific cases are transposition of verb tenses with the 'have got' expression and transposition of imperative or infinitive, which also would be ignored later.
- Input error: occurs when the text to be transposed contains errors, like missing punctuation.

Table 5.5: Precision of the Verb Tenses Transpositions - Iteration 1

From/To	FC	FGT	FPC	FPS	FS	PaC	PaPC	PaPS	PaS	PeC	PePC	PePS	PeS
						Iteration 1							
FC	-	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
FGT	0.40	-	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.40	1.00	1.00
FPC	1.00	1.00	-	1.00	1.00	1.00	1.00	1.00	.667	1.00	1.00	1.00	1.00
FPS	1.00	1.00	1.00	-	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
FS	0.80	1.00	1.00	1.00	-	1.00	0.60	1.00	1.00	1.00	0.80	1.00	1.00
PaC	0.80	1.00	1.00	1.00	1.00	-	0.80	1.00	1.00	1.00	0.80	1.00	1.00
PaPC	1.00	1.00	1.00	1.00	1.00	1.00	-	1.00	1.00	1.00	1.00	1.00	1.00
PaPS	1.00	1.00	1.00	1.00	1.00	1.00	1.00	-	1.00	1.00	1.00	1.00	1.00
PaS	0.80	1.00	1.00	1.00	1.00	1.00	1.00	1.00	-	1.00	1.00	1.00	0.40
PeC	1.00	0.80	1.00	0.80	0.60	0.80	0.80	0.60	0.80	-	0.80	0.80	0.40
PePC	1.00	1.00	1.00	1.00	1.00	0.80	0.80	0.80	0.60	1.00	-	0.80	0.60
PePS	0.80	0.00	0.40	0.60	0.40	0.60	0.60	0.40	0.40	0.60	1.00	-	1.00
PeS	0.60	0.40	0.40	0.40	0.40	0.60	0.20	0.40	0.40	0.20	0.00	0.20	-

The iteration 1 of the verb tenses transposition showed the classification errors corrected on iteration 6 of the verb tenses classification experiments.

The first transposition error corrected was the case 'he is gonna be out of a job' transposed to 'he will be carrying out of a job'. This case occurred on transpositions to the Future Continuous, which cannot be done having the verb 'be' as the main verb of the verb tense or we would have something like 'he will be being'. As this kind of structure is not used, the process took the Future continuous structure. So the transposition to the Future Continuous with 'be' as the main verb started to be routed to 'Future Simple', which is the best fit for the case of 'be' as main verb.

The kind of transposition which caused the errors classified into specific errors were blocked. Then, there will be no transpositions from infinitive form, Present Simple with 'have got' and imperative form. The verb tenses preceded by modal, like 'should have been' appeared in the first sample of the transpositions, and this kind of transposition is also blocked since they are out of the scope of the experiment. On this iteration, the modal verbs were treated for the perfect tenses.

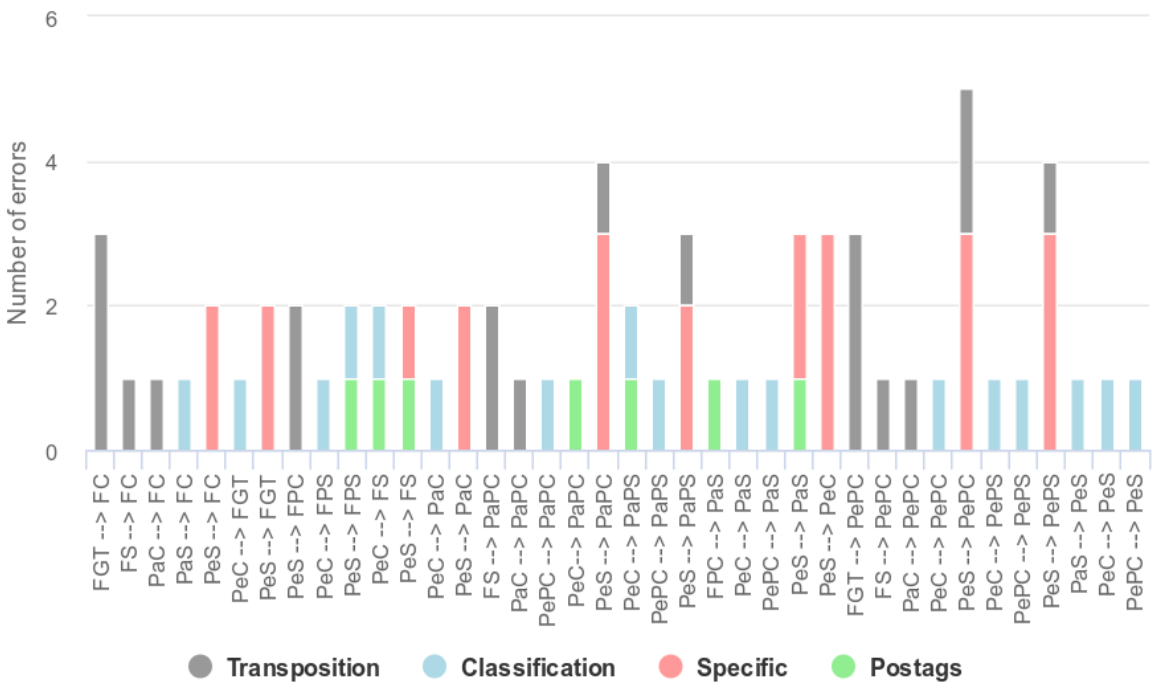

Figure 5.1: Distribution of errors for the first iteration of verb tenses transpositions

There were 26 errors of transposition of verb tenses preceded by modal. The number of the other cases of error are shown in the distribution chart in Figure 5.1. The precision of each transposition kind is shown in Table 5.5. No conjugation errors appeared in Iteration 1.

After the verb tense classification correction, the verb tenses index was rebuilt. The cases of errors were executed again. 122 transpositions were corrected, and 17 had classification errors on PoS tags.

On iteration 2, more transposition and verb tense classification errors were found.

There were 3 occurrences of the missing auxiliary verb like in the case of transposition from the Future Continuous to the Past Simple: 'you will not be shopping...' to 'you not shopped'.

There were more 17 cases of modal verb on this iteration, but this time, they happened in transposition from the Present Simple. These undesired cases were blocked.

The classification errors occurred on this iteration were described on iteration 7 of the verb tenses classification experiment. They were 8 errors.

There were 14 occurrences of transposition errors. 4 errors were caused by the Person Corrector component, where the corrected piece of text should replace the wrong one, but the wrong piece occurred in more the one place, making nonsense periods this way. The problem was corrected by adding identifiers to the pieces to be replaced. In most of the cases, the transposition process inserted unrelated and unnecessary words, because of mistakes in the process and wrong indexed rules. For instance, the sentence 'Did you have any ambition to become an actor?' in the Past Simple was transposed to 'are observing you having any ambition to become an actor?' in the Present Continuous. Other cases did not include badly inserted words like the transposition to the Past Simple 'Did you saw the papers', where there are the auxiliary verb and the main verb in the past form. The case where the participle form is used after 'will be' caused transposition error like in the sentence 'He will be dropped from...' transposed to 'He has dropped from...'. As we can see, there was a change of meaning, so, the error was treated considering the 'be' as the main verb for this case.

The person conjugation errors occurred 10 times. One error was caused by the lack of punctuation, on the try of transposing the sentence 'I guess whoever it was They were wearing gloves' missing a comma after the word 'was'. Incomplete period translations are considered errors, like in 'Did you...' being transposed to 'are doing you...'. It occurred once. There were transposition errors caused by PoS tags classification errors. In 2

Table 5.6: Precision of the Verb Tenses Transpositions - Iteration 2

From/To	FC	FGT	FPC	FPS	FS	PaC	PaPC	PaPS	PaS	PeC	PePC	PePS	PeS
						Iteration 2							
FC	-	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.80	1.00	1.00	1.00	1.00
FGT	1.00	-	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
FPC	1.00	n/d	-	1.00	n/d	1.00	1.00	n/d	n/d	1.00	n/d	n/d	n/d
FPS	1.00	0.60	1.00	-	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
FS	1.00	1.00	1.00	0.80	-	1.00	1.00	1.00	0.80	1.00	1.00	1.00	0.60
PaC	1.00	1.00	1.00	1.00	1.00	-	0.80	0.80	0.80	1.00	1.00	1.00	0.80
PaPC	1.00	0.60	1.00	1.00	1.00	1.00	-	0.80	1.00	1.00	1.00	0.80	1.00
PaPS	1.00	1.00	1.00	0.80	0.80	1.00	1.00	-	0.80	0.80	1.00	1.00	0.60
PaS	1.00	0.80	1.00	0.60	1.00	1.00	1.00	1.00	-	0.40	1.00	1.00	0.20
PeC	1.00	0.80	1.00	1.00	1.00	1.00	0.80	1.00	0.80	-	1.00	1.00	0.80
PePC	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.60	-	1.00	1.00
PePS	1.00	0.80	1.00	0.60	1.00	1.00	1.00	1.00	0.60	1.00	0.20	-	0.60
PeS	0.40	0.40	0.40	0.60	0.60	1.00	0.40	0.60	0.60	0.60	0.80	0.60	-

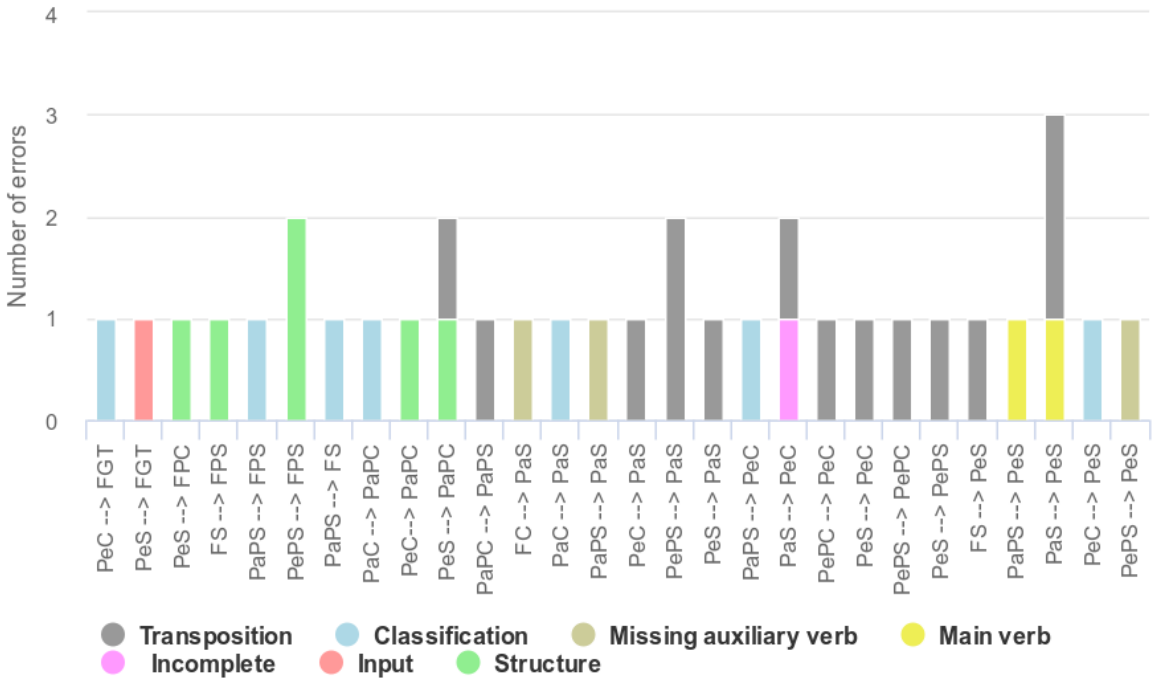

Figure 5.2: Distribution of errors for the second iteration of verb tenses transpositions

cases of transposition to the Present Simple, the verb 'be' was ignored as main verb like in the transposition 'My plan had been successful' being transposed to 'My plan have successful'. 6 structure errors occurred in transpositions of interrogative sentences to Future Perfect Simple, Past Perfect Continuous and Future Perfect Continuous. The precision of each verb tense transposition of the iteration 2 is shown on Table 5.6 and the distribution of the main errors is displayed in Figure 5.2

The table 5.7 shows the precision measures for the transpositions for iteration 3. Three new cases of classification errors occurred and they were treated in iteration 8 of the verb tenses classification experiments. On transpositions to the Past Perfect Simple, the word 'been' was inserted without need, forming sentences like 'he had been searched'. The case was corrected, so the new sentence would be 'he had searched'. This error occurred twice. Transpositions to the Past Simple would be missing the auxiliary verb 'did' for negative verb tenses, like in the period 'I had not noticed' to 'I not noticed'. The transposition was fixed to add the auxiliary verb and alter the main verb form. This error occurred once. A new case of transposition from the imperative form appeared

Table 5.7: Precision of the Verb Tenses Transpositions - Iteration 3

From/To	FC	FGT	FPC	FPS	FS	PaC	PaPC	PaPS	PaS	PeC	PePC	PePS	PeS
						Iteration 3							
FC	-	-	-	-	-	-	-	-	1.00	-	-	-	-
FGT	-	-	-	-	-	-	-	-	-	-	-	-	-
FPC	-	-	-	-	-	-	-	-	-	-	-	-	-
FPS	-	-	-	-	-	-	-	-	-	-	-	-	-
FS	-	-	-	1.00	-	-	-	-	-	-	-	-	1.00
PaC	-	-	-	-	-	-	1.00	-	0.80	-	-	-	-
PaPC	-	-	-	-	-	-	-	0.60	-	-	-	-	-
PaPS	-	-	-	1.00	1.00	-	-	-	0.80	1.00	-	-	0.80
PaS	-	-	-	-	-	-	-	-	-	1.00	-	-	0.40
PeC	-	0.80	-	-	-	-	0.80	-	1.00	-	-	-	1.00
PePC	-	-	-	-	-	-	-	-	1.00	1.00	-	-	-
PePS	-	1.00	-	0.80	-	-	-	-	-	-	1.00	-	1.00
PeS	-	-	0.60	-	-	-	1.00	-	0.80	1.00	-	1.00	-

and it was treated to be avoided. Two cases of PoS tags classification mistake occurred in iteration 3. The person conjugation errors happened 3 times.

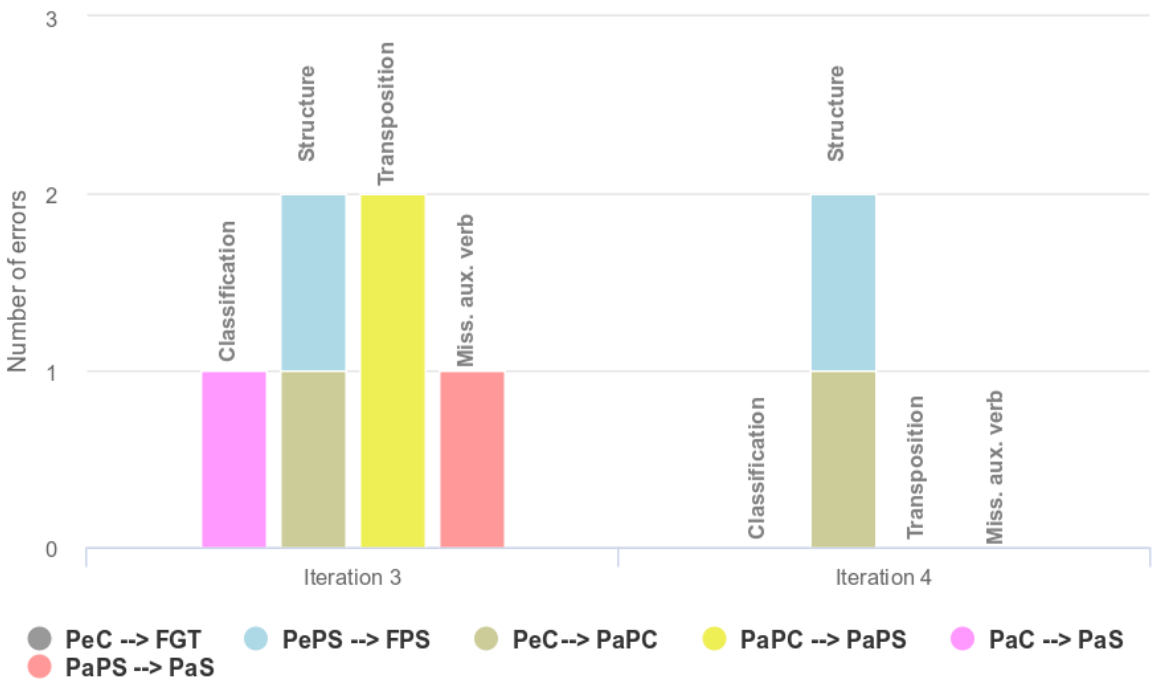

Figure 5.3: Distribution of errors for the third and fourth iteration of verb tenses transpositions

Two cases of structure error occurred, where the pronoun was misplaced like 'where had been you hiding it?' instead of 'where had you been hiding it?'.

The distribution of the errors for the iteration 3 is shown in the figure 5.3, along with the distribution of errors of the iteration 4. The experiments were needed only for 6 transpositions of verb tenses. As we can see in table 5.8, 2 transpositions remained with errors. The remaining errors are of the same kind: structure errors for interrogative sentences. The cause of the errors was not found yet and that will lower the probability of using that kind of transposition in our application.

Table 5.9 summarizes the distribution of the errors by type over the iterations. Table 5.10 summarizes the distribution of the errors in the transposition for each target verb tense over the iterations.

Table 5.8: Precision of the Verb Tenses Transpositions - Iteration 4

From/To	FGT	FPS	PaPC	PaPS	PaS
	Iteration 4				
PaC	-	-	-	-	1.00
PaPC	-	-	-	1.00	-
PaPS	-	-	-	-	1.00
PeC	1.00	-	0.80	-	-
PePS	-	0.80	-	-	-

Table 5.9: The number of errors for each error type over the iterations.

Type	1	2	3	4
Conjug.	0	10	3	0
Classif.	17	8	3	0
Struct.	0	6	2	2
Transp.	21	14	2	0
PoS tag.	8	5	2	0
Modal	26	17	0	0
Aux. verb	0	3	1	0
Main verb	0	2	0	0
Incomplete	0	1	0	0
Specific	26	0	1	0
Input	0	2	0	0
Total Err.	98	68	14	2
Success (%)	86.9	90.6	90.0	93.3
Sample Size	748	722	140	30

Table 5.10: The number of error for each target verb tense (TVT) over the iterations.

TVT	1	2	3	4
PeS	4/57	12/55	2/25	0/0
PeC	5/56	7/56	0/20	0/0
PePS	6/58	3/55	1/5	0/0
PePC	11/58	4/55	0/5	0/0
PaS	9/58	9/55	3/30	0/10
PaC	5/57	0/56	0/0	0/0
PaPS	9/57	4/55	2/5	0/5
PaPC	11/58	5/56	1/15	1/5
FS	8/58	3/55	0/5	0/0
FGT	9/57	8/55	2/10	0/5
FC	9/58	2/56	0/0	0/0
FPS	6/56	8/56	1/15	1/5
FPC	6/60	3/57	2/5	0/0
Total	98/748	68/722	14/140	2/30

5.3 Finding Verb Tense Classification Errors Using Readability

In order to find possible errors in classified verb tenses, a database has been created with each sentence containing a single verb tense, using the corpus analyzed. The saved attributes are the sentences, the verb tense window of each sentence, the readability scores of both verb tense window and sentence. The verb tense window is the sub-sentence where are contained all the words between the first and last word of that verb tense. For instance, in the sentence "I have always been studying English", the verb tense window is "have always been studying". The verb tense window has a huge set of possibilities because it accepts words with different PoS tags, like nouns, adverbs and others. A different verb tense window structure can be observed in the sentence "Have not you always been studying English?", having adverb and noun in the middle of the verb tense window. The readability score of the sentence and verb tense window is given by the language model.

After these attributes were extracted and stored, it has been selected, for each verb tense, 60 sentences with the lowest readability scores for verb tense window of the sentences. This sample of sentences should contain more errors than a sample with random selected sentences, like in the classification experiments presented in Section 5.1, because the language model is based on n-gram frequency. The lowest readability scores in the verb tenses window are related to low frequency in the corpus. Therefore, those verb tense windows in the selected sentences has a very low probability to fall in a random sample.

The verb tenses of the selected sentences were analyzed and the number of errors is shown in Table 5.11. The number shown in the first column is the raw number of errors and in the second column, the duplicates were removed. In the third and fourth columns are the respective percentages. The Future Perfect Continuous had no sentence selected, because it is rarely used and all the sentences with Future Perfect Continuous were analyzed in the classification iterations.

In all errors in the sentences, except in three, the verb tense windows had actually two verb tenses windows. Therefore, in most of the cases, the verb tenses windows were the end of a clause and the begin of another clause. The found errors in the sample are listed below.

- Specific words breaking the verb tense: It happens when there is a kind of word dividing the verb tense window. The window has actually two clauses. The splitting

Table 5.11: Misclassified verb tenses found by lowest readability scores

Verb Tense	Errors	Errors - no duplicates	Perc. errors	Perc. errors - no duplicates
PeS	29/60	29/58	48.3%	50.0%
PeC	55/60	52/57	91.7%	91.2%
PePS	3/60	3/53	5.0%	5.7%
PePC	28/60	20/45	46.7%	44.4%
PaS	31/60	22/51	51.7%	43.1%
PaC	28/60	23/51	46.7%	45.1%
PaPS	34/60	27/47	56.7%	57.4%
PaPC	6/60	5/47	10.0%	10.6%
FS	44/60	41/56	73.3%	73.2%
FGT	3/60	3/53	5.0%	5.7%
FC	12/60	11/58	20.0%	18.9%
FPS	2/60	2/44	3.3%	4.5%

words may be prepositions or "wh" pronouns. It occurred 77 times.

- Comma: splitting the verb tense window. It happened 36 times.
- Extra verb in window: The rules matches, but there are extras verbs in the middle of the verb tense window. It occurred 34 times.
- Two nouns: It occurs when the there is a noun before the verb tense window and one in the verb tense window, like in the sentences "That was two people lying" and "It will be me celebrating it", being classified as Past Continuous and Future Continuous, respectively. It occurred 27 times.
- Auxiliary verb + to be: It happened in the simple forms of Present and Past, in sentences like "All I did was just confirm those suspicions", where the verb to be is preceded by the auxiliary verbs "do" and "did". It happened 17 times.
- Gerund as infinitive: in English the gerund is also used to represent the infinitive form. This representation causes errors in the classification of the Continuous tenses. It occurred 18 times.
- Extra verb as main verb: It occurs when there is a verb after the verb tense window and it matches the rule, having the previous verb also matched the same rule. It

occurred 5 times.

- Gerund as adjective: 5 errors occurred due the gerund being used as adjective.
- Adjective: Adjective in the middle of the verb tense window indicates errors. It occurred 3 times.
- Future Continuous as Future 'Going To': sentences like "It will be going to recover the memory" were classified wrongly as Future 'Going To'.
- Expressions: 4 errors occurred due to the expressions "had better" "There will be", causing misclassification for the Past Perfect Simple and FutureContinuous, repectively.
- 2 wills: there was two cases where the modal word "will" appeared twice in the verb tense window.
- Other errors: 5 errors occurred due PoS tag classification error, lack of punctuation and bad sentence structure. 1 error occurred because the use of the participle as a noun, indicated by the preceding determiner "the".

The errors found show that low readability scores can be used to find very specific case of the natural language. The rules were under improvement to solve the new cases found.

Chapter 6

Conclusions and Future Work

In this research we developed a framework for verb tenses classification and for automatic exercises generation. The generated exercises are totally focused on verb tenses and they are ranked to learner according the readability given by the language model.

The main contributions of this research are a process for verb tenses classification in texts, and a method for transposing verb tenses. The classified verb tenses might feed features for NLP tasks, such as text categorization. On the other hand, the verb tenses transpositions might be used for text generation tasks or question generation for language learning purpose, for instance.

The framework is able to generate 4 kinds of exercises to the learner, which are verb tense transposition/transformation, fill-in-blank, exercises, true or false exercises, and multiple choice exercises.

The verb tenses classification process developed in this research is able to classify verb tenses in English texts with an average accuracy of 98.17%. The process uses a rule-based classification and it was improved over iteration. Thirteen verb tenses were included in the experiments.

Apart from the classification experiment, another experiment was performed for validating the verb tenses transposition process. The transposition process uses search and rules, and it was also improved iteratively. All the combinations of verb tenses transposition were analyzed in the experiment and for each combination of verb tense transposition (for instance, *Present Continuous to Past Perfect Continuous*) the minimum precision reached was 80%. The overall precision for the transposition process was higher than 90%.

It was shown that the low readability scores may be useful to find classification errors. In our case, it was only selected the lowest scores. However, a sample generated based on ranges of readability scores could also provide insights for the classification analysis.

The results of verb tenses classification and verb tenses transposition show that the developed framework is reliable to be used on the tasks of automatically generating exercises.

As future work we suggest three main threads. First, the effectiveness of the generated exercises applied to ESL students must be evaluated on the task of verb tenses acquisition for identifying how well the learners are progressing using the developed tool. Second, the framework is open to be extended by the inclusion of new grammar rules. Third, a user model shall be developed to fit the frameworks capabilities and the learner needs for verb tenses learning.

Data mining techniques are also an alternative for the tasks of classification and transposition of verb tenses. However, it may be hard task building or finding proper corpora for these alternatives.

Other minor points may be explored, such as the irregular form of the verbs, once it does not keep the pattern of verb conjugation and may affect the readability of a text.

Apêndice A

Appendices

A.1 Rules for the Verb Tenses Classification

This section presents the detailed rules for each of the 13 verb tenses presented on this work. The rules are combination of boolean conditions. All the boolean operators are results of analisys of a window of words and are described in the following list. The verb tense rules is shown after the list.

- matchVerb: is a function that will check if the last verb PoS tag in the window matches the needed PoS tag for the verb tense in analysis.
- doCase: This operator is used by the Present Simple and indicates wheter the current window is a case of Present Simple with the auxiliary verb "do", like in the sentece "He does not work every day".
- to, toVb: indicates the one verb in the infinitive form was found in the window in analysis.
- isModal: when the verb tense is preceded by a modal verb like "can" or "would"
- be: indicates the presence of the verb to be.
- notVbn: indicates the absence of the PoS tag VBN in the verb tense window.
- notGoingTo: indicates that the "going to" is not in the verb tense window
- jjB4Vbg: indicates that the an adjective (PoS tag "JJ") is right before the verb with PoS tag "VBG".

- initSurroundedByNouns: indicates that the begin of the rule is surrounded by nouns.
- preceddedByDT: indicates that the main verb of the verb tense is preceded by determiner.
- vbBetweenToBeAndVbg: indicates verb between the verb to be and the verb in the gerund form.
- precededByVb: indicates that the verb tense is preceded by a verb.
- vbgPrecededByFor: indicates that the verb in the gerund form is preceded by "for", "and", "of" or a word associated with the PoS tag "PRP$".
- have: indicates the presence of the verb "to have".
- vbn: indicates the presence of the verb the past participle form.
- vbBetweenVbnNVbg: indicates the presence of a verb between a verb in the participle form and a verb in the gerund form.
- noNoun: indicates the absence of nouns in the verb tense.
- didCase: indicates the formation of the Past Simple with the use of the "did" auxiliary.
- notHave: indicates the absence of the verb "to have".
- goingTo: indicates the presence of the "goingTo".
- vbgBetweenVbgNBe: indicates the presence of a verb in the gerund form between a verb "to be" and a gerund verb.
- had: indicates the presence of the verb "have" in the past form
- notVbg: indicates the absence of a verb in the gerund form.
- beAfterHad: indicates that the verb "to be" is present after the "had" verb.
- will: indicates the presence of the auxiliary "will".
- noVerbBetweenWillNVerb: indicates the absence of a verb between the auxiliary "will" and the main verb of the verb tense.

- going: indicates the presence of the verb “go” in the gerund form.
- vb: indicates presence of a verb.
- noPronouBetweenHaveNVerb: indicates the absence of pronouns between the verb “to have” and the main verb of the verb tense.
- noVerbBetweenWillNHave: indicates the absence of verb between the auxiliary “will” and the verb “have”.
- vbnIsRightB4Vbg: indicates that the verb in participle form is before the verb in the gerund form.

Present Simple Rule
(matchVerb(“VB(P|Z)?”) OR doCase) AND !to AND !isModal

Present Continuous Rule
matchVerb(“VBG”) AND be AND notVbn AND notGoingTo AND !to AND !jjB4Vbg AND !initSurroundedByNouns AND !preceddedByDT AND !vbBetweenToBeAndVbg AND !precededByVb AND !vbgPrecededByFor

Present Perfect Simple
matchVerb(“VB(N|D)?”) AND have AND noNoun AND !toVb

Present Perfect Continuous
matchVerb(“VBG”) AND have AND vbn AND !jjB4Vbg AND !toVb AND !vbBetweenVbnNVbg AND !preceddedByDT

Past Simple
matchVerb(“VBD”) OR didCase

Past Continuous
matchVerb(“VBG”) AND be AND notVbn AND notHave AND !goingTo AND !jjB4Vbg AND !vbgBetweenVbgNBe AND !initSurroundedByNouns AND !preceddedByDT AND !precededByVb AND !vbgPrecededByFor

Past Perfect Simple

matchVerb("VB(N|D)?") AND had AND notVbg AND noNoun AND !toVb AND !beAfterHad AND !initSurroundedByNouns

Past Perfect Continuous

matchVerb("VBG") AND had AND vbn AND !jjB4Vbg AND !toVb AND !vbBetweenVbnNVbg AND !preceddedByDT

Future Simple

matchVerb("VB(P)?") AND will AND noVerbBetweenWillNVerb

Future Going To

matchVerb("VB(P)?") AND going AND to AND vb

Future Continuous

matchVerb("VBG") AND will AND be AND !jjB4Vbg AND !preceddedByDT AND !vbgPrecededByFor

Future Perfect Simple

matchVerb("VB(N|D)?") AND will AND have AND notVbg AND noPronounBetweenHaveNVerb AND noVerbBetweenWillNHave AND noNoun AND !toVb

Future Perfect Continuous

matchVerb("VBG") AND will AND have AND vbn AND noVerbBetweenWillNHave AND vbnIsRightB4Vbg AND !jjB4Vbg AND !toVb AND !preceddedByDT AND noNoun

Bibliography

Anagnostou, N. K. and Weir, G. R. (2006). From corpus-based collocation frequencies to readability measure, *In ICT in the Analysis, Teaching and Learning of Languages, Preprints of the ICTATLL Workshop 2006*, pp. 33–46.

Barzilay, R. and Lapata, M. (2008). Modeling local coherence: An entity-based approach, *Computational Linguistics* **34**(1): 1–34.

Beinborn, L., Zesch, T. and Gurevych, I. (2012). Towards fine-grained readability measures for self-directed language learning, *Proceedings of the SLTC 2012 workshop on NLP for CALL; Lund; 25th October; 2012*, number 080, Linköping University Electronic Press, pp. 11–19.

Bögel, T., Strötgen, J. and Gertz, M. (2014). Computational narratology: Extracting tense clusters from narrative texts, *In Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, pp. 950–955.

Bormuth, J. R. (1969). Development of readability analysis.

Brown, J. C., Frishkoff, G. A. and Eskenazi, M. (2005). Automatic question generation for vocabulary assessment, *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, pp. 819–826.

Brown, J. D. (1997). An efl readability index, *University of Hawai'i Working Papers in English as a Second Language 15 (2)* .

Callan, J. and Eskenazi, M. (2007). Combining lexical and grammatical features to improve readability measures for first and second language texts, *In Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Rochester, United States of America, pp. 460–467.

Coleman, M. and Liau, T. L. (1975). A computer readability formula designed for machine scoring., *Journal of Applied Psychology* **60**(2): 283.

Collins-Thompson, K. (2014). Computational assessment of text readability: A survey of current and future research, *International Journal of Applied Linguistics* **165**(2): 97–135.

Collins-Thompson, K. and Callan, J. (2005). Predicting reading difficulty with statistical language models, *Journal of the American Society for Information Science and Technology* **56**(13): 1448–1462.

Dale, E. and Chall, J. S. (1948). A formula for predicting readability: Instructions, *Educational research bulletin* pp. 37–54.

Dale, E. and Chall, J. S. (1949). The concept of readability, *Elementary English* **26**(1): 19–26.

Derczynski, L. and Gaizauskas, R. J. (2015). Temporal relation classification using a model of tense and aspect., *RANLP*, pp. 118–122.

Duman, G., Orhon, G. and Gedik, N. (2015). Research trends in mobile assisted language learning from 2000 to 2012, *ReCALL* **27**(02): 197–216.

Feng, L., Jansche, M., Huenerfauth, M. and Elhadad, N. (2010). A comparison of features for automatic readability assessment, *In Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, Association for Computational Linguistics, Beijing, China, pp. 276–284.

Flesch, R. (1948). A new readability yardstick., *Journal of applied psychology* **32**(3): 221.

Greenfield, G. R. (1999). *Classic Readability Formulas in an EFL Context: Are They Valid for Japanese Speakers?*

Greenfield, J. (2004). Readability formulas for efl, *JALT Journal* **26**(1): 5–24.

Gunning, R. (1952). The technique of clear writing.

Heilman, M., Collins-Thompson, K., Callan, J. and Eskenazi, M. (2006). Classroom success of an intelligent tutoring system for lexical practice and reading comprehension., *INTERSPEECH*.

Heilman, M., Collins-Thompson, K. and Eskenazi, M. (2008). An analysis of statistical models and features for reading difficulty prediction, *In Proceedings of the Third Workshop on Innovative Use of NLP for Building Educational Applications*, Association for Computational Linguistics, Columbus, United States of America, pp. 71–79.

Heilman, M. and Smith, N. A. (2010). Good question! statistical ranking for question generation, *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Association for Computational Linguistics, pp. 609–617.

Huang, Y.-T., Chang, H.-P., Sun, Y. and Chen, M. C. (2011). A robust estimation scheme of reading difficulty for second language learners, *Advanced Learning Technologies (ICALT), 2011 11th IEEE International Conference on*, IEEE, pp. 58–62.

Kidwell, P., Lebanon, G. and Collins-Thompson, K. (2011). Statistical estimation of word acquisition with application to readability prediction, *Journal of the American Statistical Association* **106**(493): 21–30.

Kılıçkaya, F. (2015). Computer-based grammar instruction in an efl context: improving the effectiveness of teaching adverbial clauses, *Computer Assisted Language Learning* **28**(4): 325–340.

Kincaid, J. P., Fishburne Jr, R. P., Rogers, R. L. and Chissom, B. S. (1975). Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel, *Technical report*, DTIC Document.

Kitson, H. D. (2010). *Mind of the Buyer*, Nabu Press.

Lee, J. (2011). Verb tense generation, *Procedia-Social and Behavioral Sciences* **27**: 122–130.

Lively, B. A. and Pressey, S. L. (1923). *A method for measuring the" vocabulary burden" of textbooks.*

Loaiciga, S., Meyer, T. and Popescu-Belis, A. (2014). English-french verb phrase alignment in europarl for tense translation modeling, *The Ninth Language Resources and Evaluation Conference*, number EPFL-CONF-198442.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S. and McClosky, D. (2014). The stanford corenlp natural language processing toolkit., *ACL (System Demonstrations)*, pp. 55–60.

McLaughlin, G. H. (1969). Smog grading: A new readability formula, *Journal of reading* **12**(8): 639–646.

McLaughlin, G. H. (1974). Temptations of the flesch, *Instructional Science* **2**(4): 367–383.

Rabab ah, G. A. and AbuSeileek, A. F. (2009). The effect of computer-based grammar instruction on the acquisition of verb tenses in an efl context.

Schwarm, S. E. and Ostendorf, M. (2005). Reading level assessment using support vector machines and statistical language models, *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, Association for Computational Linguistics, pp. 523–530.

Si, L. and Callan, J. (2001). A statistical model for scientific readability, *In Proceedings of the tenth international Conference on Information and Knowledge Management*, ACM, Atlanta, Georgia, pp. 574–576.

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

Thorndike, E. L. (1927). *The teacher's word book*, Teachers College, Columbia Univ.

Tiedemann, J. (2012). Parallel data, tools and interfaces in opus., *LREC*, pp. 2214–2218.

Vajjala, S. and Meurers, D. (2012). On improving the accuracy of readability classification using insights from second language acquisition, *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, Association for Computational Linguistics, pp. 163–173.

Wang, H. X. (2016). Developing and testing readability measurements for second language learners.

Xia, M., Kochmar, E. and Briscoe, T. (2016). Text readability assessment for second language learners, *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 12–22.

Young, R. (2011). *How Audiences Decide: A Cognitive Approach to Business Communication*, Taylor & Francis.