Computational Linguistics: Analysis of The Functional Use of Microsoft Text Word Processor Text Corrector

Priscilla Chantal Duarte Silva
Prof. Dr. Ricardo Luiz Perez Teixeira
Victoria Olivia Araujo Vilas Boas
Federal University of Itajubá, Brazil

Doi: 10.19044/llc.v6no2a2 URL:http://dx.doi.org/10.19044/llc.v6no2a2

Abstract

Computational linguistics is a field of study that lies at the interface between Linguistics and Computer Science. Though, it is an area that lacks the cooperation of both areas of knowledge as well as other areas of the Cognitive Sciences. The field of Computational Linguistics has the posturing of attending to Computation with regard to the treatment of linguistic data, analyzing the approach and the application of the computational components that try to reproduce the natural language phenomenon. The present study aims to show the advance of computational linguistics, its motivations, applications, as well as the relation with the natural language, comparing the form of application of the computation with the linguistic functioning of the language of Portuguese Language. The study proposes a linguistic analysis of the text correctors with the approach of its limitations and inaccuracies compared to the natural language. Some phrases in the mother tongue were selected and inserted into Word as tests to fix possible grammatical errors and to check the current text corrector limitations, for later analysis and collection of results. The results revealed that Microsoft Word language reviser can not correct all Portuguese language errors. This indicates that it is necessary to review the conditions and operations of the Microsfot Word reviser engine.

Keywords: Computational Linguistics, Text Corrector, Natural language, Natural language processing.

Introduction

The process of architecting characteristics of the human being to the machine has undergone some transformations over the years. When, in 1950, Alan Turing - a mathematician, cryptanalyst and British computer scientist -

proposed to the scientific community, for the first time, a thinking machine, in its article "Computing Machinery and Intelligence", researchers in the area directed works in search of expressing humans in bytes. However, over the years they have not made much progress, and therefore have segmented the approach. Today, the study of artificial intelligence is subdivided into areas such as computer vision, voice analysis and synthesis, fuzzy logic, artificial neural networks, computational linguistics, and the like.

Computational linguistics, even as a subdivision of Artificial Intelligence - in addition to areas such as Statistics, Linguistics and Information Technology - precedes these studies, for in the mid-1950s Americans tried to automatically translate documents written in other languages in order to speed up the work to process information they obtained from, for example, spies infiltrated into Soviet environments. At the time, the computer was gaining strength in calculating complex mathematical expressions, such as the precise routes of airplanes and rockets launched by NASA; if the algebraic calculations had a precision that goes beyond human efforts, the same could apply to natural languages such as English, Russian, German, and others.

However, although the range of automatic translators of the time had a small result, it was not perfect and therefore the computational scientist community understood that there is a great complexity in the treatment of natural languages and began to devote greater efforts to the creation of algorithms and software capable of doing this work. Good and Howland (2017) argue whether natural language might be a preferred notation to traditional programming languages, given its familiarity and ubiquity. They describe and destille empirical studies investigating the use of natural language for computation and the ways in which different notations, including natural language, can best support the various activities that comprise programming.Today, computational linguistics is subdivided into some areas such as: corpus linguistics, syntactic analysis, part-of-speech tagging, knowledge representation, information retrieval, semantic web and machine translation.

One of the most popular applications is the automatic correction in word processors like LibreOffice Writer, Microsoft Word and Apple Pages. They are used for writing simple texts to professional and complex files; simulate a typewriter, but also have tools that aid in textual production, formatting, and editing. However, none of today's word processors and brokers are completely efficient, and this is what drives computer scientists and linguists to continue their research in an attempt to develop the perfect tool whose ability is to syntactically and semantically correct a text with property, resembling human thought. Some softwares are being developed to meet people's daily need for writing.

The most efficient do not encompass the needs of the Portuguese-speaking writer, since they tend to restrict themselves in the English language. Grammarly, of the large company Grammarly, located in San Francisco, for example, is a word processor that proposes a greater efficiency in the correction of writing in the English language; is the most popular software in North America, being developed about six years ago - it can be used as an extension of Microsoft's Word (the most popular processor in the world) and of browsers like Google Chrome. In addition, the Language Tool - another example of a textual correction tool - was developed about 10 years ago, and DeepGrammar about a year ago. However, of the above textual correction tools, only the Language Tool is available in the Portuguese language, and even promising to go beyond Word, you can not detect some of the errors that will be exposed in this article.

Even with the emergence of word processors that promise greater efficiency in dealing with natural languages through newer and more innovative technologies, this study has as its main focus the analysis of Word by addressing its limitations and inaccuracies when it comes to to the vast amount of information - syntactic, semantic, morphological that need to be known for the rescue of the signifier. After all, there is a real difficulty on the part of linguists and engineers of language in articulating, understanding, processing speech and writing. These are characteristics of the human being, and even if scientists believed it to be easy to reproduce, the practice of systematizing this human function confers its complexity. For the complete interpretation of sentences of a given language it is necessary to have a kind of knowledge of syntactic rules of a sentence, which according to the linguist Noam Chomsky in his book Language and Mind (2006) is intrinsically formed in the speaker's mind - he they have somehow internalized - and this is the ability to associate sounds and meanings according to rules of their mother tongue. Thus, textual correction faces the same difficulty, and therefore tends to be imprecise.

Knowing that the human mind has a peculiar organization in its way of processing ideas, inferring terms and decoding information, it is understood that the process of rectifying texts is not based on only a comparison of right and wrong, but a procedure of understanding that is not done with excellence by existing platforms. Artificial intelligence, in the area of Deep Learning - AI sphere that proposes the deep learning of the machine through the elaboration of neural networks to compose the layers of unnatural thinking - has shown many results in researches and in the creation of modern word processors. Thus, the considerations made by some linguists regarding the intrinsic knowledge of language by human beings will also be approached in the present article, in the search to understand about the inefficiency of the

available brokers in Portuguese language, and how Deep Learning can prove useful for the Portuguese in relation to the orthographic correction.

The general objective of this work is to analyze linguistically the comprehension dimensions of text correctors in dealing with the mother tongue, verifying their limitations and inconsistencies. For this, it will be necessary to: raise a historical panorama on the advances of the artificial intelligence in the field of text correctors; to get an overview of the contributions of computational linguistics in the area of Artificial Intelligence, to understand the mechanism of computational and linguistic functioning of the Microsoft Word broker - because it is the most widely used around the world - investigating syntactic verification means, semantic and morphological aspects of this software based on grammatical and semantic inaccuracies. Finally, present and analyze the results, pointing out suggestions for improvement.

2. Computational Linguistics: a general overview

Our main research is conducted on the basis of one IT company, which simultaneously develops equipment, software, and also provides a range of services to its customers. The main business of the company is the development of satellite-based monitoring system for different types of customer objects. Computational Linguistics consists of an area of knowledge that explores the relationships between Linguistics and Informatics (Vieira and Lima, 2001), in an attempt to formulate systems capable of recognizing and producing information in natural language. Deprecating the operation of the rules of a language and, of course, what allows recognizing the system of all others is the challenge of computational linguistics, in order to approach the formal language of the natural. According to Vieria and Lima (2001), some works in Computational Linguistics are focused on the processing of natural language. For this, it is necessary to understand the structural functioning of the language. Lingusitic processing is the task of the syntactic parsers so that they recognize the lexicon and grammar of a language. It is known that the syntax of natural language is much more complex than any form of formal processing.

Yang et al. (2017) identify in their ontogenesis of child language some evidences of learning mechanisms and principles of efficient computation, i.e, that children make use of hierarchically ('Merge') language. Edelman (2017) suggests that the brain learning mechanisms remain dynamically controlled constrained navigation in concrete or abstract situation spaces. Love (2017) speculate how languaging about language might give rise to the idea of a language. The author observes the role of reflexivity and the development of writing in facilitating the decontextualisation, abstraction and reification of linguistic units and languages themselves.

Normally, a native speaker is able to recognize a sequence of expressions as valid in their language. This is because there is a set of internalized rules, which Chomsky (2006) has classified as part of the formal functioning of language or the formal nature of language. It is known that there is a complexity in dealing with the approach and systematization of this area of knowledge that makes it difficult to find a uniformity in theses. In this respect, (Câmara Junior, 1973, p. 50, apud Alkmin T. M, p.23) argues that according to Schleicher, each language is the product of the action of a complex of natural substances in the brain and in the speech apparatus. Studying a language is therefore an indirect approach to this complex of matters[...] he argued that language is the most appropriate criterion for the rational classification of humanity.

Within this conception, the study of a language encompasses notions that are sometimes foreign to science and often pervade philosophical debates. What is proposed by the author bears great resemblance to the work and defense of the linguist, philosopher and political activist Noam Chomsky; the famous debate with Michel Foucault, a French philosopher, in the year 1971 illustrates the above disagreements: For Chomsky, contrary to Foucault, human nature does not change essentially in the different cultures and historical periods, since humans have characteristics correlated to rudimentary existence. Chomsky (2006) argues that there is a difference in each culture and period of history that does not allow one to speak in an immutable human nature, or in an innate species. Chomsky (2006), in turn, emphasizes the creativity of to illustrate the process of language learning by children; argues that it is not limited to the performance of external agents. In Linguagem e Mente, the linguist claims that the study of Natural Languages - a term referring to what is naturally developed by the human being, such as the Portuguese Language, English Language, and the like - is directly related to the human essence and to the qualities of the mind that are unique to man and independent of phases or factors of life.

Chomsky (2006) defends the idea that in general sentences have an intrinsic meaning determined by a system of rules internalized by the speaker of a language. However, it stresses that they are not just connections between sound and meaning. In other words, it is not only a matter of interpreting what is said from the application of linguistic principles that determine phonetics the semantic properties of an utterance, but believes that extralinguistic factors confer on the speaker the role of determining how the language is produced, identified and understood. Linguistic performance is governed by principles of a cognitive structure.

The grammar of a language consists of a cognitive model composed of a set of pairs (s, I), where s is the phonetic representation of a certain linguistic sign and I is the semantic interpretation. There is, in fact, a perceptual model

that can be described as a sign that functions as an input and allows for syntactic, semantic, and phonetic representation. In this sense, this perceptual model incorporates the grammar of a language. Chomsky (2006) emphasizes, in turn, that the perceptual model makes use of much information that lies behind the intrinsic association between sound and meaning. The grammar of a language involves issues of memory, time and organization of perceptual strategies that go beyond formal grammar. Considering the grammar of a language in this perceptual model, one can better understand the notion of universal grammar proposed by Chomsky (2006), which consists of a system of representation that serves any particular language, although it considers the arbitrariness of the linguistic sign, in the sense that a language has an infinity of signs of semantic interpretation.

For Chomsky (2006), the grammar of a language is a system of rules that comprise a pair of sound and meaning, surrounded by a syntactic, semantic and phonological component. This is the formal aspect of language. The proposed method for personalized e-learning can be described within further sta The syntactic component is a certain infinite class of abstract objects (D, S), where D is a deep structure and S is a surface structure. The deep structure contains all information relevant to the semantic interpretation; the surface structure all information for the phonetic interpretation. The semantic and phonological components are purely interpretive (Chomsky, 2006, p.111). The structure of the syntactic component of a grammar contains certain network of grammatical functions followed by rules or system of rules of the deep structure combined with rules of the surface structure. In Computational Linguistics, syntactic parse.

In Computational Linguistics, syntactic parsers are able to recognize and validate a sequence of expressions as being of a given language. In this case, it is necessary to specify a grammar. In this case, it is necessary to specify a grammar accompanied by its lexicon so that the system performs the checks. According to Vieira and Lima (2001), the procedure is similar to checking the syntax of a program in a programming language. In natural language, the system is much more complex. This type of treatment is useful for the development of spelling and grammar correctors. The applications developed to deal with language, however, go beyond syntactic processing, considering the postulations of Chomsky (2006), there is a frustration that goes through the study of Linguistics, related to the complexity of human language. After all, even with all the progress of studies and approaches, sometimes the systematizations or reproductions of languages run into the same dilemmas of the Human Language. Thus, for an efficient approach to language elements, linguistics being the field of scientific study of language and natural languages was, over time, formulated on campus that shaped a deeper understanding for the analysis of phenomena.

The areas of morphological, phonological, syntactic, semantic, pragmatic and historical are the most important in the science of language. The morphological one is concerned with the study of the composition of words through the minimal units that carry meaning; the phonological studies the signifier of the language, the phonic differences related to the differences of meaning; the syntactic field that is, among others, most frequently addressed, refers to the way words are related in the search for sentence composition. Semantics, in turn, deals with the signifier of the meaning of words and phrases; already the pragmatic one tries to understand the motivation of the interlocutor in the construction of the speech. Finally, historical linguistics studies the processes of language transformation throughout history. What we are suggesting is that the notion of "understanding a sentence" is explained in part in terms of the notion of "linguistic level." To understand a sentence, then, it is first necessary to reconstruct its analysis at each linguistic level; and we can test the appropriateness of a given set of abstract linguistic levels by ascertaining whether the grammars formulated in terms of these levels allow us to provide a satisfactory analysis of the notion of "understanding" or not (Chomsky, 1956, p.81).

For a long time, the scientific community has focused attention on syntax and since the 19th century there has been a great desire to establish it as an autonomous discipline; even by standardizing her study, differences in approaches have emerged. Two main lines of thought related to language stand out: the functionalist and the formalist. The first sees language as a system that is born of the individual's need for communication. The excerpt from the book Introduction to Linguistics clarifies the explanation: to think of syntax from a functionalist perspective implies, then, to extend the analysis beyond the limits of the sentence. Syntactic processes are understood here by the relations that the syntactic component of the language maintains with the semantic and discursive components. It is only possible to understand what is happening in Syntax, also looking at the context (text and / or communicative situation) in which the sentence is inserted. (Chomsky 2004, 212)

The second line of thought concerns the formal aspect of language. That is, the formalists argue questions related to the linguistic structure, and approaches are given, in this case, the sentence and its structuring. The great systematiser of formalism was Noam Chomsky; by his work, this linguistic theory became known as Transformational Grammar or Generative Grammar. It was a priori introduced in the work of the linguist "Synthetic Structures" in the 1950s, and its main purpose was to provide a general method of selecting a grammar for each language given a corpus of phrases of that particular language. The ultimate goal, therefore, was an ideal model

for all known languages. In view of this, this branch of linguistics was a great attraction for mathematicians and computer scientists, as they sought, at the same time, a formulation and computational processing of language. In other words, in an attempt to speed up the translation of Russian scientific papers after the launch of Sputnik in 1957, for example, the researchers imagined that syntactic transformations based on Russian and English grammars and the substitution of words with the use of an electronic dictionary would be sufficient to preserve the exact meanings of utterances. Different languages have different morphological tendencies. Computational methods of analysis that are perfectly suitable for languages of morphological shortage (such as English), or with agglutination morphology (such as Turkish), may not be the best methods for non-flexions languages (such as Russian) (Ledeneva, Y; Sidorov, G. 2010, p.5).

The fact is that translation requires general knowledge of the subject to resolve ambiguities and establish the content of the sentence - the famous translation of "the spirit is willing but the flesh is weak" as "vodka is good, but the flesh is rotten" illustrates the difficulties encountered. If indeed there is a precise formulation of all languages, then reproducing it in algorithms generates, as a consequence, efficient processing.

In this way, Computational Linguistics arises; according to the Association for Computational Linguistics, or ACL - a scientific and professional society dedicated to the problems involved in the study of language from the computational point of view - is the scientific study of language from a computational perspective that is interested in providing computational models for the various types of linguistic phenomena, with the aim of reproducing them in computational processes in the treatment of natural languages.

The area deals with applications that, even with all the advances over the 70 years of research, are still difficult to reproduce. After all, if we start from Chomsky's (2006) assumption that there are human abilities related to language that are intrinsic to his existence, then how to reproduce them in a machine? For this reproduction to be efficient is, therefore, necessary to integrate a set of human characteristics into the machine? In order to answer questions such as these, the various fronts of computational linguistics - speech recognition, speech synthesis, search engines, machine translation, automatic correction and word processing, extracting text information and automatic summarization - deal with semantic and encompass questions such as "Is a given solution the best solution?" or "Given any program, is this program correct?", classic computer theory questions.

That is, to appropriate resolution requires the ability to process information and act in decision making on top of what has been processed. This goes beyond the use of a grammatical corpus for the statistical mapping

of possibilities, although this is one way of doing it. However, the problem sometimes involves human characteristics, such as the capacity to think. In the field of automatic translation of texts, this has meant over the years of research: In the most recent systems competitions promoted by the National Institute of Standards and Technology (NIST, 2008), the best automatic translation system (Google) did not even reach 50% of the human reference. [...] None of the available systems, derived from the market initiatives, are derived from the academic research, produced until today, results that could dispense with human edition (Martins, 2011, page 287).

On top of very syntactic analyzes, the Theory of Computing encompasses finite automata; recognizers of regular languages that work as follows: Given any language and taking a sequence of x elements that compose it, an automaton responds "yes" if it encounters x and "no" if it comes across any other sequence not known. That is, it works on aspects of that language that are known and therefore do not make decisions. In general, they are programmed to deal with comparison instructions that follow pre-established standards and encompass concepts that are responsible for defining what is grammatically acceptable to a native speaker and what he / she is not. Note that to establish grammar goals significantly, it is sufficient to assume a partial knowledge of sentences and non-sentences. That is, we will admit, in this discussion, that certain phoneme sequences are clearly sentences and that other sequences are not. In many intermediate cases, we will have no hesitation in leaving the decision to the grammar itself, when it is constructed in the simplest form, so as to include clear sentences and exclude sequences that are clearly non-sentences (Chomsky 1993, 8).

The semantic questions mentioned in the previous paragraph require that the word intelligence be reproduced in its literal sense. By the dictionary Aurelio (2016), the general definition of intelligence is: "Set of all the intellectual faculties (memory, imagination, judgment, reasoning, abstraction and conception)"; in other definitions, the word learning is also present. This means that for a semantic understanding there are elements that are directly related to the mental faculties, and issues that, at times, the human being himself has no conclusions. A concrete example of the above is the MIT platform called Moral Machine, which reproduces a stand-alone car in a decision-making situation. The vehicle warns the user that an accident will occur at some point and asks for help in deciding which people to take their lives - a group of children, or a group of elderly people; thugs or doctors and children.

In modern times, machines and tools designed from computational linguistics are very useful in people's daily lives. Some tools make up daily work and study routines, such as word processors; other tools, has been the focus of research on applicability. The chatbot - a chat robot that simulates a

human being in a conversation - is an example of the second case and has been used in a variety of applications; language courses, shops, classes, etc. After all, the tendency of computing is to facilitate the routine of people in the streamlining of processes that once demanded time and work. The focus of this study is the approach of word processors, especially Microsoft's Word, and its various limitations - syntactic factors, semantic factors, etc. For this, computational linguistics can work with a textual corpus that has probabilistic models based on the most adequate grammar of a given language. The ability to produce and recognize grammatical statements is not based on notions of statistical approximation and the like. The custom of considering as grammatical sentences those that "may occur" or that are "possible" has been responsible for some confusion ... I believe that we must consider that grammar is autonomous and independent of meaning and that probabilistic models do not provide some insight into some of the basic problems of syntactic structure (Chomsky, 1998, p.11, 12).

This for the author can be a mistake because using a corpus to statistically map the most possible sentences leads to the thought that language, because of its complexity, can not be properly described, and then there must be contentment to a schematized version, making probability low - a certain word has a low frequency of use - impossible. The statistical study of language has relevance, but it can not determine or characterize the set of grammatical statements. For non-inflected languages this method is highly suitable; others, however, need algorithms responsible for processing them properly. One extreme point is storing all grammatical forms in a dictionary (database). Such method of analysis is known as "bag of words". This method is useful for inflective languages, but it is not recommended for agglutinative or polysynthetic ones […] Algorithmic (non- "bag-of-words") solutions have a number of additional advantages. For example, suck algorithms have the possibility to recognize unknown (new) words. This is a crucial feature for a morphological analyzer since new words constantly appear in the languages, not speaking of possible incompleteness of the dictionary (Ledeneva, Y, Sidorov, G. 2010, p.6).

3. Computational processes involving the processing of a text

Automatic text processors are part of everyday digital users. The most used to write long texts - contrary to those found in smartphone operating systems and the like - is Microsoft's Word. This tool, among many other functions, works with "Grammar checking" and "Style checking". The first concerns the checking of errors that are usually handled by a grammar book: syntax errors stand out. The second deals with errors discussed by books about writing style: we can take as an example the exacerbated use of "que", called “queísmo” in Portuguese or the exaggeration in the use of the passive voice.

The treatment of problems that a usual grammar addresses and discusses is best handled by word processors such as Word. This is because it concerns a kind of comparison made between user input and a kind of library of possible structures, similar to the use of compilers in computer programming - text style checking, in turn, involves semantic problems, which as discussed in the previous chapter, are difficult to interpret by the machine. Thus, for a better approach to the computational processes that involve the automatic processing of text, a brief introduction to the use of the compilers is necessary. Put simply, a compiler is a program that reads a program written in a language - the source language - and translates it into an equivalent program in another language, the target language. As an important part of this translation process, the compiler reports to its user the presence of errors in the source program (Aho, A. Sethi, R., Ullman, J., 1995, 8).

In general, the compilation of a program is divided into its central parts: analysis and synthesis. In the first part of the process, the structure and meanings of the program will be explored from the creation of intermediate representations for the verification of errors. In the analysis, the program will recognize if character structuring has meaning for the grammar of a given language, a process called "sentence recognition" and is part of the lexical analysis procedure. Thus, if a particular programming language defines an integer as int or "integer" - for example, Fortran -, the declaration of a variable of the integer type must be accompanied by an integer.

Otherwise, the compiler will convert the variable to the declared type (a real number 5.4895, will be understood as 5). If the declaration of a numeric type - in the chosen language: real, integer or complex - is accompanied by a string, called a string, the compiler will acknowledge an error, because when declaring a number, a character is not expected.

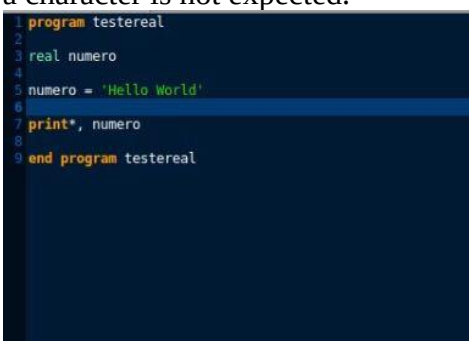

Fig.1- Using the Compiler Program Code.

As the code example, a real variable was created, and a string was assigned to it, so the compiler says that the conversion is impossible.

Fig. 2 - Description of errors.

The compilation, according to the above-mentioned book Compilers - Principles, Techniques and Tools (Aho, A. Sethi, R., Ullman, J.,1995), is divided into three parts: Linear Analysis, Hierarchical Analysis and Semantic Analysis. Linear Analysis, also known as Lexical Analysis, is responsible for the detailed reading of the program - character to character -, and its separation into tokens, which are its minimum structures; equivalent in Linguistics to words; they carry the signifier of each element. The '+' symbol, for example, can be a token that carries addition direction, just as '=' can mean assignment.

Hierarchical Analysis, or commonly called Syntactic Analysis, associates tokens in usually syntactic trees for the summarization of outputs and for the evaluation of the disposition of these elements in sentences - in linguistics, syntax is the part of the grammar that studies the association of the words in a sentence and the acceptability of the formal relations that interconnect them. The semantic analysis, in turn, verifies errors of semantic order in the code. A classic example of an error identified at this stage is the division of a number by zero, considered a mathematical indeterminacy; or the division of an integer by a floating point number.

A basic principle of compilation, and therefore essential for the understanding of word processing, is the classification of formal grammars described by Chomsky in 1959. For a better understanding of what these grammars are, it is necessary to elucidate some concepts and structures that Chomsky expounded in his book Synthetic Structures (1957). As a simple example of the new form of grammar associated with the analysis of constituents, consider the following:

(13)
(i) Sentence -> SN + VP
(ii) SN -> T + N
(iii) SV -> Verb + SN
(iv) T -> o, a
(v) N -> man, ball, etc.
(vi) verb -> kicked, caught, etc.
[...] where the numbers to the right of each line of the derivation refer to the rule of the "grammar" (13) used to construct the line from a preceding line.
(Chomsky, 1957, p.20)

Thus, to form the sentence "the man kicked the ball", Chomsky (1956) follows the logic of formation, starting from that the numbers to the right of each line of the derivation refer to the rule of the one of the declared grammar above. In this way:

(14) Judgment
SN + SV (i)
T + N + SV (ii)
T + N + Verb + SV (iii)
or + N + Verb + SN (iv)
the + man + Word + SN (v)
the + man + kicked + SN (vi)
the + man + kicked + T + N (vii)
the man + kicked the N (viii)
the man kicked the ball

Thus, the second line is (14) is formed from the first line by the rewrite of the Sentence as SN + SV, according to rule (i) in (13); the third line is formed from the second line by the rewriting of SN as T + N, according to rule (ii) of (13), etc. We can represent derivation (1) in an obvious way through the following diagram:

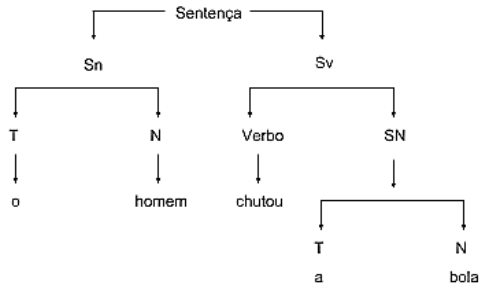

Fig. 3 – Chomsky diagram
Chomsky (1957, p. 20, 21).

Thus, if a sequence is the last, it can be said that it is a terminal derivative. This concept is essential for the understanding of the terminal and non-terminal symbols that form the formal grammars. That is, a non-terminal symbol corresponds to the range of symbols upon which grammar rests to define laws for the composition of sentences of that language. In contrast, terminal symbols are the ultimate derivation of grammar and cannot be altered by its rules. In programming languages they are Tokens and are usually treated as synonyms so that they can be manipulated by syntactic parsers that provide their proper meanings - as previously discussed.

This diagram became known as the Chomsky tree, and its explanation as Noam Chomsky's Generative Theory. From understanding these syntactic trees it becomes simple to understand the classification of grammars that he theorized years later. Chomsky hierarchized the formal grammars into 4 levels - from level 0 to level 3 - with zero level grammars being more general, or with more level of freedom in their norms. The Type 0 grammar, where alpha -> beta (any number of variables or terminals can occur in any order within a production) is called unrestricted and is capable of generating recursively enumerable languages, which are formal languages for which they exist turing machines - abstract model of a computer, which is restricted only to the logical aspects of its operation (memory, states and transitions) - to enumerate all the valid language chairs in order to perform steps recursively an arbitrary number of times. Because it is a more general grammar, the type 0 has limitations as to the applicability to the compilers, because there are great obstacles regarding its treatment due to the generality of its composition.

A subset of sentence structure grammars are context-sensitive grammars, or type 1 grammars can be surrounded by a terminal symbol context and non-terminal symbol on the left or right side of their production rule - the alpha side and the (alpha -> beta) both terminal variables and non - terminal variables are allowed, but the alpha side must necessarily be less than or equal to the beta side - and are ordered enough to be verified by a Turing machine with limited memory .

Subsequently, context-free grammars are subsets of the others mentioned above; they were invented by Chomsky in the search for the processing and comprehension of any natural language through the propositions that govern them with the description of the structure of the sentences and words, but were not originally used for this purpose, having later great applicability in the science of computation, to the description of programming languages and, recently, the creation of the eXtensible Markup Language (XML) recommendation for the generation of markup languages by the World Wide Web Consortium (W3C). The concept of context-free languages was of paramount importance in order for XML to achieve its goal of being a simple and readable language, both for humans and for computers.

The last subset of Chomsky's hierarchy contains the regular grammars, which are responsible for the constraints on the forms of production and, therefore, are simple and adequate to obtain recognizers, also called regular expressions, of great applicability in computer science and therefore in the development of software. A common example of using regular expressions is the validation of email by forms available on the network to the standard example@example.com:

^([A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+.[A-Za-z]{2,4}){0,1}$

The first part of the expression, for example, validates that the user can type letters A through Z, uppercase and lowercase, numbers from 0 to 9; the characters "underline", percent, more, and dash. Almost all human languages, or natural languages, source of study for the correction and for the automatic translation of texts are part of the set of context-sensitive grammars. The treatment of these grammars is within the set of problems of complexity and computation theory called decision problem, in which it is verified whether or not a given string belongs to a set of characters called formal language; this question is answered with yes or no. This is important for a proper understanding of the operation of a word processor such as Microsoft's word, because in a simple way, Microsoft's natural language processing system parses in essentially the same way the languages of at the time of compilation - described earlier. In addition, they work with a database for comparison, where the decision-making complexity method fits as in the following example: Is this word written to fit any of the available records? Yes or no.

The NLP system that is behind the Microsoft grammar checker is a full-fledged natural language processing system that is also intended to be used for many other applications. It consists of a programming language and a runtime environment that are both specially tailored to the needs of an NLP system. The programming language, which is called G, has basically the syntactic appearance of the C language, but it gives special notational support to attribute-value data structures, called records, and provides an additional programming construct, called rules. The runtime system, which is usually referred to as NLPWin, is a Microsoft Windows application that is written mostly in C and provides a grammar development environment and the functions needed to do natural language processing. That part is the processing that is written in G, such as the English analysis grammar, is translated into C by a program called Gtran, and then it is compiled and linked into the NLPWin executable (Dale, R; Moisl, H; Somers, H. 2000, p.182)

In general, there is a set of predetermined records - such as a database for consultation - a set of rules also predetermined and a method of grammar analysis, which considers in complexity of decision making if a given test is correct or not; if it is not, it will accuse, as in the compilers. This analysis is done in six stages. The first stage of processing is lexical, where the input text is segmented into individual tokens, which are primarily words and punctuation marks. [...] The second stage of processing is called the syntactic sketch, and corresponds to what is typically called parsing. [...] The third stage is called the syntactic portrait, because it is a refinement of the syntactic sketch. The purpose of this processing is to produce more reasonable attachments for some modifiers, such as prepositional phrases and relative clauses that are merely attached to their closest possible modificand as a simplification in sketch. [...] The fourth stage of processing produces logical

forms, which are intended to make explicit the underlying semantics of the input text. [...] The fifth stage of processing deals with lexical disambiguation [determining the most appropriate sense (or senses) of each word in the input text]. [...] The sixth, and final, stage of processing deals with discourse phenomena. These six stages can be compared to the three fundamental parts of analysis made by the compilers described above: Linear, Hierarchical and Semantic. The first stage described can be compared to Linear Linear Analysis, according to the fourth stage it fits into the Hierarchical Analysis, and the last stage in the Semantic Analysis. The description of the Analysis made by word will be of great value for the analysis of some sentences grammatically incorrect; tested in Word, but that he did not give them any kind of error.

4. Methodology

The present study has a bibliographic and descriptive bias. Therefore, at first, a bibliographic review was carried out. According to Cervo et al (2006) the bibliographic research is one that tries to explain a problematic from the use of already published theoretical references. This research can be carried out independently or and analyze the scientific contributions on the topic in question. When this research is carried out with the purpose of making a survey of previous information and knowledge in relation to a problem for which answers are sought, or in relation to a hypothesis that one wishes to try, this type of research composes part of the descriptive research, or experimental. In a second moment, some phrases in the mother tongue were selected and inserted into Word as tests to fix possible grammatical errors and to check the current text corrector limitations, for later analysis and collection of results.

5. Systemic analysis of word processor behavior on grammatically incorrect sentences

From the analysis of some linguistic tests done in Word with the purpose of identifying grammatical errors that are not pointed out, it was possible to obtain a list of grammatical mistakes in Portuguese that are easily identified by an attentive reader. Some of them, such as problems of verbal agreement, use of commas and crass, semantics, anaphora.

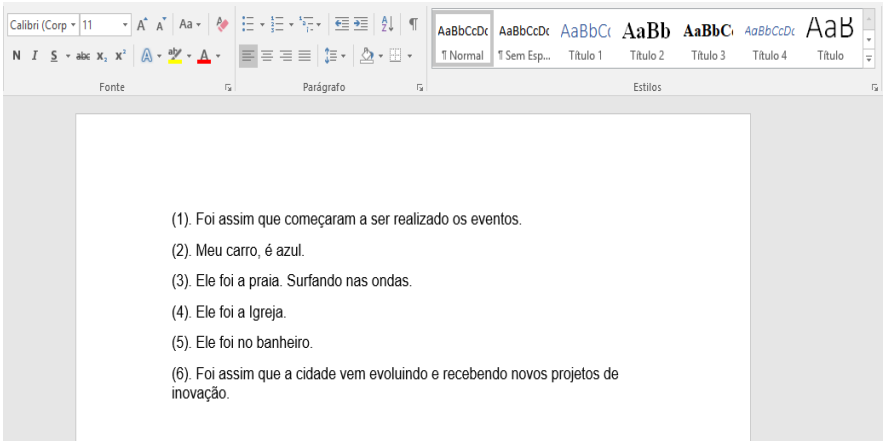

Fig. 4 - Examples of grammatical errors that Word does not identify as an error

This is how the events began to take place, (2) My car, is blue, (3) He went to the beach. Surfing the waves, (4) He went to church., (5) He went in the bathroom, (6) This is how the city has been evolving and receiving new innovation projects.

The non-corrective IDs of the Word text corrector will be explained at the time of the systemic linguistic analysis of the broker's functioning, with a relation to Compilers of Computing languages. In example (1) "*This is how the events began to take place*", there is a problem of verbal agreement, which was not identified by the Word broker. In a reversal of order of terms, one can observe the inadequacy of grammatical rules: This is how the events began to be performed. Note, in this case, that the broker cannot identify issues that run away from the order subject, verb and predicate. In some cases, one observes only an attempt of agreement between nouns and adjuncts like in: "Many accidents happened". The forms "happened many accidents"; "Accident happened a lot"; "Accidents happened a lot."

Note, therefore, a non-compliance with grammatical rules. The same happens in: "Buy want the students"; "Students buy them want"; the students want to buy ", in which semantics is not identified. In (2), "My car is blue", the use of the comma after the subject, separating predicate subject does not follow the grammar rules of punctuation. The same thing happens in "My car is. Blue ", in which the lack of the predicate of the subject is not identified; nor the predicative of the object "The teacher. left. John disconsolate ". just like "They hit. Hours on the Clock ", where there is a comma-separated verb and direct object. In (3), "He went to the beach. Surfing in the waves ", the absence of the crass is not marked, as in (4) and there is still a misused comma separating the sentences with semantic link. It is believed that there is a problem in identifying semantic and structural elements. In (5), "He was in the bathroom," the use of the regency "ao" is also not identified, accepting the

wording with the preposition "no." In (6), there is no concern with the use of semantically connected verbal time with the use of gerund, which in this case gives idea of continuity. It is noted, therefore, that Microsoft's Word broker is only returning to verify spelling indicia without pragmatic linkage, that is, its contextual and semantic articulation. There is every reason to believe that this type of broker should be programmed for a more complete recognition of the language, involving more notions of grammatical and semantic order.

The work developed sought the systemic analysis of the use of word corrector word. For this to be possible, it was necessary to clarify all the topics of the area of Computational Linguistics that surround the processing of text done by the automatic correction of Word. Firstly, an introduction to Linguistics in general, from basic concepts to the most complex ones, and that provide the basis for the understanding of other areas that were important for this research; secondly Computational Linguistics - guiding all work -, which gives base to all the works involving the computational processing of sentences; thirdly it was necessary to understand how the previously learned concepts in the two mentioned parts are important so that the machine is able to gauge the ability to correct a text.

From this, it was possible to make the analysis of some sentences for a more systematic understanding of how a program such as word, - with all the research that involves it and any advances that it has obtained over the last 5 to 7 years - can fail. This is because the human mind has a peculiar organization in its way of processing ideas, inferring terms and decoding information, and it is understood that the process of rectifying texts is not based on just a comparison of right and wrong or a certain input of data compared to a pre-established base of expected inputs, but a procedure of understanding data that a machine has not yet achieved complete success and which resembles the process performed by the human brain.

Artificial intelligence, in the area of Deep Learning - AI sphere that proposes the deep learning of the machine through the elaboration of neural networks to compose the layers of unnatural thinking - has shown many results in researches and in the creation of modern word processors. One case studied for this work was Jonathan Mugan, a computer scientist living in Texas: DeepGrammar. This textual broker uses deep learning to create language models, and these models are used to analyze the test in a few steps, which consider the semantic meanings of words.

This method proved to be a great success, and in a year of work the researcher alone achieved results close to the results presented by word in more than 10 years of research and with a great team of researchers. The analysis made by the system becomes, in this case, more similar to human analysis; This is because, for a human being to correct in a speech a wrong word, he must first know the meaning of that word.

6. Conclusion

From this study it was possible to understand how Microsoft Word reviser works. The reviser presents some limitations of verbal agreement, use of commas and crass, problems of semantic order and of anaphora. The results show that many corrections made by Word's text editor are not done correctly. It is possible to observe that the corrector of texts makes only corrections of grammatical nature

In fact, there needs to be a review of the functions of the Word broker. This may be an opportunity for future work in computational linguistics and also for computer engineering itself. After all, the software should be able to do the correction in multiple languages. It was noticed that the correction made by Word focuses more on the spelling correction. Semantics is still a more difficult field because it concerns the human capacity to think.

References:

1. Aho, A. Sethi, R., Ullman, J. (1995). Compiladores: Princípios, Técnicas e Ferramentas. Rio de Janeiro: LTC, 8.
2. Angelo, T. N. (2011). Behaviorismo Radical e Inteligência Artificial: Contribuições além das Ciências Cognitivas. Faculdade de Engenharia Elétrica e Computação da Unicamp. Disponível em: <http://www.dca.fee.unicamp.br/~gudwin/courses/IA889/2011/IA889-19 >Acesso em 15 set. 2016.
3. Chomsky, N. (1956). Estruturas Sintáticas. Cambridge: Department of Modern Languages and Research Laboratory of Eletronics.
4. Chomsky, N. (1998). Linguagem e Mente. Brasília: Editora da Universidade Federal de Brasília.
5. Chomsky, N. (2006). Language and mind. New York: Cambridge.
6. Dale, R., Moisl, H., Somers, H., 2000. Handbook of Natural Language Processing. New York: Marcel Dekker.
7. Delmonto, R. (2008). Computational Linguistic Text Processing: Lexicon, Grammar, Parcing and Anaphora Resolution. New York: Nova Science Publishers, Inc., 375. – ISBN 978-1-60456-749-6
8. Edelman, S. (2017). Language and other complex behaviors: unifying characteristics, computational models, neural mechanisms. Language Sciences, 62, 91-123.
9. Gardner, H. (2003). A Nova Ciência da Mente. São Paulo: Edusp.
10. Good, J., & Howland, K. (2017). Programming language, natural language? Supporting the diverse computational activities of novice programmers. Journal of Visual Languages & Computing, 39, 78-92.
11. Ledeneva, Y, Sidorov, G. (2000). Recent Advances in Computational Linguistic. Informativa, México. Disponível em:

<http://www.informatica.si/index.php/informatica/article/view/271/>. Acesso em: 25 set. 2017.
12. Love, N. (2017). On languaging and languages. Language Sciences, 61(6), 113-147.
13. Martins, R. (2011). O pecado original da Linguística Computacional. São Paulo: Alfa, 287.
14. Mira, M., Villalva, A. (2006). O Essencial Sobre a Linguística. Lisboa: Caminho.
15. Mussalim, F.(Org.), Bentes, A. C. (2004). Introdução à linguística. São Paulo: Cortez, 23.
16. Norvig, P. (1992). Paradigms of artificial intelligence programming: case studies in common lisp. San Francisco: Elsevier, 655-680.
17. Riezler, S. (2014). On the Problem of Theoretical Terms in Empirical Computational Linguistics. MIT Press Journals, Computational Linguistic, vol.40, n.1, Cambridge, 235-245.
18. Russell, S.; Norvig, P (2004). Inteligência artificial. Tradução de PubliCare Consultoria. Rio de Janeiro: Elsevier, 9a reimpressão.
19. Silva, B. C. D. da. (2006). O estudo Linguístico-Computacional da Linguagem. Porto Alegre: Letras de Hoje, 2006.
20. Yang, C., Crain, S., Berwick, R. C., Chomsky, N., & Bolhuis, J. J. (2017). The growth of language: Universal Grammar, experience, and principles of computation. Neuroscience & Biobehavioral Reviews