Natural language processing for social inclusion: a text simplification architecture for different literacy levels

Caroline Gasperin[1], **Erick Maziero**[1], **Lucia Specia**[1], **Thiago Pardo**[1], **Sandra M. Aluisio**[1]

[1]NILC - Núcleo Interinstitucional de Linguística Computacional
ICMC, Universidade de São Paulo
Av. Trabalhador São-Carlense, 400 - 13560-970 - São Carlos/SP, Brazil

{cgasperin,lspecia,taspardo,sandra}@icmc.usp.br, egmaziero@gmail.com

Abstract. *Text simplification is a research area of Natural Language Processing, whose goal is to maximize text comprehension through simplification of its linguistic structure. This paper presents our approach for Brazilian Portuguese text simplification. As people have different literacy levels, we take that into account when generating simplified texts. We propose an architecture for text simplification composed by two layers: the first is a machine-learning system who learns from manually simplified texts the appropriate degree of simplification according to a given literacy level; and the second is a rule-based system that executes the actual simplification of the sentences, following the recommendations from the first layer.*

Resumo. *A Simplificação Textual é uma área de pesquisa do Processamento de Língua Natural cujo objetivo é maximizar a compreensão de textos escritos via simplificação de sua estrutura linguística. Este artigo apresenta nossa abordagem para simplificação de textos em português do Brasil. Como as pessoas possuem níveis diferentes de letramento, levamos isso em consideração na geração de textos simplificados. Propomos uma arquitetura para simplificação de textos composta de dois níveis: o primeiro é um sistema baseado em aprendizado de máquina que aprende a partir de textos simplificados manualmente o nível apropriado de simplificação de acordo com um dado nível de letramento; e o segundo é um sistema baseado em regras que executa a simplificação propriamente dita das sentenças, seguindo recomendações vindas do primeiro nível.*

1. Introduction

IBGE’s 2006 Summary of Social Indicators [IBGE 2006] shows that in 2005 Brazil had around 14.9 million illiterate people who were 15 or older (11% of the population), according to data provided by PNAD (National Household Sample Survey) for that year. If we add to this number people with less than four years of schooling (called functional illiterates) the proportion of the population increases significantly: 23.5%.

In order to gather more detailed and reliable information related to these statistics the INAF index (National Indicator of Functional Literacy) was created. According to INAF’s 5-year report [INAF 2007], a vast number of people belong to the so called rudimentary and basic literacy levels. These people are only able to find explicit information in short texts (rudimentary level) and also process slightly longer texts and make simple inferences (basic level). The INAF report showed that 68% of the 30.6 million Brazilians between 15 and 64 years who have studied up to 4 years remain at the rudimentary literacy level, and 75% of the 31.1 million who studied up to 8 years remain at the rudimentary or basic levels.

It is well-known that long sentences, conjoined sentences, embedded clauses, passive voice, non-canonical word order, use of low-frequency words increase sentence complexity and make texts difficult to read for people at poor literacy levels [Klebanov et al. 2004, Devlin and Unthank 2006, Siddharthan 2003]. The PorSimples project (*Simplificação Textual do Português para Inclusão e Acessibilidade Digital*[1]) aims at producing text simplification tools for promoting digital inclusion and accessibility for people at such levels of literacy, and possibly other kinds of reading disabilities. Our focus is on texts published in government sites or by relevant news agencies, both expected to be of importance to a large audience with various literacy levels. The language of the texts is Brazilian Portuguese, for which there are no text simplification systems, to the best of our knowledge.

The outcome of our research can help promote universal access to knowledge among the Brazilian population, which is one of the four great challenges of computer science in Brazil for the coming years [SBC 2006].

Text simplification is a research area of Natural Language Processing, whose goal is to maximize text comprehension through simplification of its linguistic structure. The simplification can involve the substitution of words that are only understood by a small group of people for common words, and the change in the syntactic structure of the sentences. As a result, the text is expected to be easily understood by people or computational systems. Text simplification has been exploited in other languages for helping poor literacy readers [Max 2006, Siddharthan 2003], bilingual readers [Petersen and Ostendorf 2007] and special kinds of readers such as aphasics [Devlin and Unthank 2006] and deaf people [Inui et al. 2003]. It has also been used for improving the accuracy of other natural language processing tasks [Chandrasekar and Srinivas 1997, Klebanov et al. 2004, Vickrey and Koller 2008], like parsing and information extraction.

As people have different literacy levels, in this paper we present an architecture for text simplification that is able to accommodate that. Our architecture is composed by two layers: the first is a machine-learning system who learns from manually simplified texts the appropriate degree of simplification according to a given literacy level; and the second is a rule-based system that executes the actual simplification of the sentences, following the recommendations from the first layer. We also present here our first experiments on implementing both layers of our architecture.

Our text simplification architecture will be the core of two online applications: a web browser plug-in for online simplification of texts on the Web, and an authoring tool to help writers to create simplified texts.

In the next section, we describe existing work on text simplification that relate to ours. In Section 3 we give an overview of our architecture and give more details about the degrees of simplifications that we consider. In Section 4 we detail the process of creating the corpus of natural and strong simplifications and the resulting corpus, from which we

[1] http://caravelas.icmc.usp.br/wiki/index.php/Principal

extract the examples to train the first layer of the architecture, presented in Section 5. In Section 6 we present the second layer of our architecture.

2. Related work

To the best of our knowledge, there is no text simplification system that aims to provide varying degrees of simplification according to the user needs. Moreover, none of the existing systems address the language under consideration in this paper, Brazilian Portuguese, for which the need of text simplification is evident, given the number of poor literacy readers, as mentioned in Section 1. Although many of the simplification operations apply across different languages, some are specific or at least more relevant for certain languages. The actual simplification system is also language-dependent, given that rules are usually defined based on linguistic features. Therefore, it would not be possible to apply existing systems to Portuguese.

Existing text simplification systems can be compared along three axes: the type of system – rule-based or corpus-based –, the type of knowledge used to identify the need for simplification, and the goals of the system.

A few rule-based systems have been developed for text simplification [Chandrasekar et al. 1996, Siddharthan 2003], focusing on different readers (poor literate, aphasic, etc). These systems contain a set of manually created simplification rules that are applied to each sentence. These are usually based on parser structures and limited to certain simplification operations. Siddharthan's approach uses a three-stage pipelined architecture for syntactic text simplification: analysis, transformation and regeneration. In his approach, POS-tagging and noun chunking are used in the analysis stage, followed by pattern-matching for known templates which can be simplified. The transformation stage sequentially applies seven handcrafted rules for simplifying conjoined clauses (coordinating, subordinating and correlative), relative clauses and appositives. The regeneration stage of the system fixes mistakes introduced by the previous phase by generating referring expressions, selecting determiners, and generally preserving discourse structure with the goal of improving cohesion of the resulting text.

Corpus-based systems, on the other hand, can learn from corpus the relevant simplification operations and also the necessary degree of the simplification for a given task. The study that is closest to ours is that by [Petersen and Ostendorf 2007], but their goal is different: learning the rules governing the simplification in order to inform second language teachers. They adopt machine learning techniques in order to learn when to drop a sentence from the text and when to split a sentence. For splitting sentences, a C4.5 classifier is trained using 20 features (shallow, morphological and syntactic ones). An average error rate of 29% is obtained in this classification task. The lengths of sentence and noun phrase were found to be the most important features.

3. Architecture overview

To attend the needs of people with different levels of literacy, we propose two types of simplification: *natural* and *strong*. In our project, the first type is aimed at people at the basic literacy level and the second at the rudimentary level. The difference between these two is the degree of application of simplification operations on the sentences.

For strong simplification we apply a set of pre-defined simplification operations to make the sentence as simple as possible, while for natural simplification certain simplification operations such as splitting and inversion of clause ordering are applied with parsimony. Two of the operations – putting the sentence in its canonical order (subject-verb-object) and putting the sentence in the active voice – are always applied since they are understood as general guidelines. This "naturalness" is based on a group of factors, which are difficult to define using hand-crafted rules, therefore we intend to learn them from examples of natural simplifications.

We propose an architecture that can handle both degrees of simplification. It is composed by two layers: the first is a machine-learning system who learns from manually simplified texts when to apply simplification operations to a sentence so that the resulting simplified text is considered natural; the second is a rule-based system that implements all simplification operations and executes them when recommended by the first layer. For strong simplification, the text only needs to pass by the second layer. Figure 1 presents our architecture.

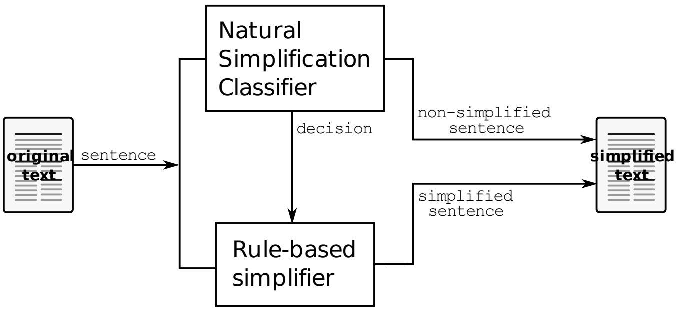

Figure 1. Simplification process

For natural simplification, each sentence of the original text passes by the first layer, natural simplification classifier, and if it decides that the sentence should be simplified, the sentence proceeds to the second layer, where the simplification actually occurs, otherwise it is left untouched. For strong simplification, where all sentences should be simplified, they go straight to the second layer.

3.1. Natural vs. strong simplification

Table 1 shows examples of an original text from an on-line Brazilian newspaper in (A), its natural simplification in (B) and its strong simplification in (C). The first sentence in (B) can be further simplified if split in shorter ones, as shown in (C). (C) however may look somehow redundant, but it can be useful for people with very low literacy levels [Williams and Reiter 2005].

The gradation between natural and strong simplification has an educational character since understanding and learning through texts are not enhanced when based only on simple texts [Ramos 2006]. Although simplification is an educational action that teachers perform on a daily basis, this action must be well balanced to improve students' learning skills. We expect our simplification tools to be used in the educational setting – in our project we also want to help poor literacy people to improve their reading skills over time.

Table 1. An example of an original text (A) and its simplified versions (B and C)

A	A taxa de inadimplência das pessoas físicas subiu em janeiro pelo quarto mês seguido e alcançou o maior patamar desde maio de 2002. A alta foi puxada principalmente pelas linhas de financiamento de veículos, que fecharam o mês passado com a maior inadimplência da série histórica do BC, iniciada em 1991.
B	A taxa de inadimplência das pessoas físicas subiu em janeiro pelo quarto mês seguido e alcançou o maior patamar desde maio de 2002. As linhas de financiamento de veículos puxaram a alta principalmente. As linhas de financiamento fecharam o mês passado com a maior inadimplência da série histórica do Banco Central, iniciada em 1991.
C	A taxa de inadimplência das pessoas físicas subiu em janeiro pelo quarto mês seguido. A taxa de inadimplência alcançou o maior patamar desde maio de 2002. As linhas de financiamento de veículos puxaram a alta principalmente. As linhas de financiamento fecharam o mês passado com a maior inadimplência da série histórica do Banco Central. A série histórica do Banco Central foi iniciada em 1991.

3.2. Implementation

We have started implementing both layers of our architecture and have done some initial experiments. For the first layer, we have trained a binary classifier to decide whether a sentence should be split or not in order to automatically obtain a natural simplified text – sentence splitting is the most important and most frequent syntactic simplification operation, so we decided to focus on this operation first. This experiment is detailed in Section 5.

For the second layer, we have implemented simplification rules for the most complex syntactic constructs according to the Manual of Syntactic Simplification for Portuguese also developed in the project [Specia et al. 2008, Aluísio et al. 2008]. The implementation of this layer is detailed in Section 6.

4. Corpus creation

In order to get a better understanding of the simplification task and to build training and evaluation data sets, we have built a corpus of manually simplified texts.

The corpus of texts chosen to be annotated with its simplifications was extracted from one of the main Brazilian newspapers, *Zero Hora*. We developed a tool to assist human annotators in this inherently manual task – the Simplification Annotation Editor[2]. We also propose a new schema for representing the original-simplified information, based on the XCES standard[3]. The annotation tool and corpus encoding is detailed in [Caseli et al. 2009]. The parallel corpora resulting from the simplification process can be queried in a public Portal of Parallel Corpora of Simplified Texts[4].

The Simplification Annotation Editor, besides facilitating the manual simplification process, records the simplification operations made by the annotator. The Editor has two modes to assist the human annotator: the Léxico and the Sintático modes. In the *Léxico* mode, the editor proposes changes in words and discourse markers by simpler

[2] http://caravelas.icmc.usp.br/anotador/

[3] http://www.xml-ces.org

[4] http://caravelas.icmc.usp.br/portal/index.php

and/or more frequent ones. The *Sintático* mode proposes syntactic operations based on syntactic clues provided by a parser for Portuguese [Bick 2000]. When the annotator selects an operation, it is recorded and the annotator can specify what has been changed in the simplified version.

The Simplification Annotation Editor follows a 3-step architecture. In the first step, the original text is created (or simply opened from a file). In the second step, natural simplifications are produced and from these, strong simplifications are generated (step3).

Our general simplification guidelines encourage shortening of sentences, canonical order and passive to active voice transformation. The set of lexical and syntactic simplification operations defined in our project and that can be applied to a sentence in the original text is the following: (1) non-simplification; (2) simple rewriting (replacing discourse markers or sets of words, like idioms or collocations) or (3) strong rewriting (any sort of free rewriting of sentence, as defined in [Petersen and Ostendorf 2007]); (4) putting the sentence in its canonical order (subject-verb-object); (5) putting the sentence in the active voice; (6) inverting the clause ordering; (7) splitting or (8) joining sentences; (9) dropping the sentence or (10) dropping parts of the sentence, and (11) lexical substitution. The lexical operations are (11) and (2), which consist of replacing uncommon words or longer expressions by simpler synonyms, respectively.

When performing a natural simplification, the annotator is free to choose which operations to use, among the 11 available, and when to use them. There are cases where the annotator decides not to simplify a sentence. Strong simplification, on the other hand, is driven by explicit rules from a Manual of Syntactic Simplification for Portuguese, which state when and how to apply the simplification operations, with the goal of simplifying the text as much as possible.

4.1. The parallel corpus of original and simplified texts

The resulting annotated corpus is composed of 104 news articles. Table 2 shows the total number of sentences and words and the average sentence length (in words) of the original, natural and strong simplified versions of the texts. A considerable reduction can be observed with respect to individual sentence lengths from original to simplified texts, which is a consequence of splitting sentences. The overall text length is longer than the original, which was expected, since simplification usually yields the repetition of information in different sentences, particularly when splitting operations are performed.

Table 2. Statistics on the original, natural and strong corpora

	Original	Natural	Strong
# Sentences	2,116	3,104	3,537
# Words	41,897	43,013	43,676
Average sent. length	19.8	13.85	12.35

Tables 3 and 4 show the number and percentage of sentences with respect to the input texts after the simplifications from original (O) to natural (N), and from natural to strong (S), focusing on two aspects: the types of operations applied and the syntactic phenomena addressed. In Table 3, most operations can be combined and applied to the same sentence, except the “Non-simplification” and “Dropping sentence” operations, which are exclusive. In the natural simplification process, the most common operation is lexical substitution, followed by splitting sentences. Strong simplifications (from natural simplifications) prioritize splitting sentences and lexical simplification (lexical substitution and simple rewriting). The higher number of non-simplification operations in the strong simplification process is due to the fact that most of the sentences had already been simplified in the natural simplification phrase.

Table 3. Statistics on the simplification operations

Simplification Operations	Number of sentences / % O → N		N → S	
Non-simplification	418	19.75	2,220	71.52
Strong rewriting	7	0.33	4	0.13
Simple rewriting	509	24.05	313	10.0
Subject-verb-object ordering	31	1.46	13	0.42
Transformation to active voice	89	4.21	65	2.09
Inversion of clause ordering	191	9.03	74	2.38
Splitting sentences	723	34.17	380	12.24
Joining sentences	5	0.24	6	0.19
Dropping sentence	6	0.28	3	0.09
Dropping sentence parts	241	11.39	49	1.58
Lexical Substitution	980	46.31	196	6.34

Table 4. Statistics on the syntactic phenomena

Syntactic Phenomena	Number of sentences / % O → N		N → S	
Apposition	196	9.26	54	1.74
Coordinate Clause	806	38.09	801	25.80
Passive Voice	198	9.35	146	4.70
Relative Clause	521	24.62	412	13.27
Subordinate Clause	452	21.36	524	16.88

As shown in Table 4, certain syntactic phenomena are more frequent than others, and therefore a larger number of simplification operations on sentences containing those types of phenomena were applied. The most frequent ones are coordinate, relative and subordinate clauses. These are in general the most difficult cases to simplify, according to studies performed in our project. Deciding whether to split a sentence containing multiple clauses is a problem on its own, since it is not always possible to divide such sentences and in many cases, even if it is possible, the resulting sentences would not read well (see examples in Section 5.2. These are additional motivations for the construction of corpus-based tools to support the simplification process.

5. First Layer: Natural simplification machine-learning system

The first layer of our architecture consists of a classifier which is trained on a set of examples from our corpus of natural simplified texts in order to learn when to simplify a sentence in order to obtain a natural simplified text. In order to learn when to apply each of the simplification operations, it would be necessary to train one classifier per operation based on examples of that operation.

At the moment, we have only experimented with the sentence splitting operation, which is the most frequent syntactic simplification operation, and can be seen as a key point of distinction between natural and strong simplification, as shown in Table 3. A binary classifier is trained with a large number of features to identify which sentences should be split in order to produce a natural simplified text, as described in what follows.

5.1. Feature set

From the analysis of our annotated corpus, we extract a number of features which aim to describe the characteristics of the sentences involved (or not) in splitting operations. Table 5 lists our feature set, which includes superficial, morphological, syntactic and discourse-related features. Features 1 to 26 are considered our *basic* feature set. We consider them as such because they reflect the findings of previous work and also the findings of our own work within the project, that is, they encode characteristics that are known to influence the complexity of the sentences and consequently its suitability for simplification. Features 2 and 4-18 are similar to the ones proposed by [Petersen and Ostendorf 2007]. The remaining features are based on lexicalized cue phrases (27 to 183), which include conjunctions and discourse markers such as “assim” and “ao invés de”, and rhetoric relations (184 to 209) (associated with sets of cue phrases) such as “conclusion” and “contrast”. [Williams 2004] has discussed the use of cue phrases in the context of language simplification. The cue phrases and rhetorical relations used here are derived from the ones produced by a discourse analyzer for Brazilian Portuguese [Pardo and Nunes 2006].

Table 5. Feature set

#	Feature	#	Feature
1	number of characters	15	average size of PPs
2	number of words	16	number of VPs
3	average size of words	17	average size of VPs
4	number of nouns	18	number of clauses
5	number of proper names	19	number of coordinated clauses
6	number of pronouns	20	number of subordinated clauses
7	number of verbs	21	number of relative clauses
8	number of adjectives	22	is there an appositive clause?
9	number of adverbs	23	is the sentence in passive voice?
10	number of coordinative conjunctions	24	number of cue phrases
11	number of subordinative conjunctions	25	is there a cue phrase in the beginning of the sentence?
12	number of NPs	26	is there a cue phrase in the middle of the sentence?
13	average size of NPs	27-183	number of occurrences for each cue phrase of a list (157 cue phrases)
14	number of PPs	184-209	is there a rhetoric relation *x* present in the sentence? (26 rhetoric relations)

Since the cue phrases and rhetorical relations are usually very sparse, we have applied different feature selection methods in order to select which of the corresponding features are relevant to our problem. We have experimented with a few known feature selection algorithms implemented in Weka [Witten and Frank 2005] (Information Gain, Wrapper, Principal Components, among others) but these have not helped improving the performance on the classification task. We have then adopted a simpler feature selection strategy, we trained classifiers using one feature at a time and all features except one, and selected all features that performed above average in the first case and below average in the second case. From this set, we added the n best performing features to the basic set, testing different values of n. The best results on a validation set were obtained with the basic set of features plus the top 50 performing features.

5.2. Classification

In order to learn whether to split or not a sentence for natural simplification, we have trained a classifier on the manually annotated corpus. Each sentence in the corpus is represented by a given set of features. Sentences are tagged as positive instances if they were annotated as containing a splitting operation; otherwise they are tagged as negative.

We use Weka's SMO implementation of Support Vector Machines (SVM) as classification algorithm, with radial basis function kernel and optimized complexity and gamma parameters. Our initial dataset contains 728 examples of the splitting operation and 1328 examples of unsplit sentences, which we randomly split in one subset for training (75%) and one for testing (25%). We actually create five different 75%-25% training-test random splits in order to have a better performance estimate. The training sets are further split into validation-train (70%) and validation-test (30%) sets for parameter optimization and feature selection. We report the average performance on these five test splits. In Table 6 we present the results of the classification task using four different feature sets: (1) the feature set used by [Petersen and Ostendorf 2007], (2) our *basic* set, (3) all our features, and (4) our *basic* set plus the best 50 additional features.

Table 6. Results using different feature sets

Feature set	Precision	Recall	F-measure
Petersen	71.68	71.54	71.58
Basic	72.48	72.34	72.34
All	72.56	72.48	72.46
Basic+50	**73.50**	**73.42**	**73.40**

Considering [Petersen and Ostendorf 2007]'s features as our baseline, we show that the features that were added to this baseline yielded a slight increase in the performance of the classifier. The addition of all the discourse-related features contributed to a further small increase in performance. Nevertheless, adding only the top 50 discourse-related features showed considerable improvement with respect to the baseline features.

If we compare our results with a simpler baseline, a classifier which always choose the majority class, we observe a greater improvement. Such classifier would obtain Precision of 40.0%, Recall of 63.3% and F-measure of 49.1%.

Table 7 shows an example that our classifier decided to split and another that it did not; we show the original and simplified (natural) versions of these examples. The sentence splitting operation is executed when the sentence contains apposition, relative clauses, coordinate or subordinate clauses. Sentence (1) in Table 7 was chosen by our classifier to be split, while sentence (2) was chosen not to be split. Both sentences contain relative clauses, but sentence (2) is not a good candidate for splitting because the main clause would become meaningless without the relative clause. We can observe factors that have influenced the correct classification of both sentences (1) and (2): the difference in sentence length, the higher number of clauses and phrases in (1), the longer phrases in (1), the presence/absence of discourse markers.

Table 7. Split and non-split sentences

1	O	Ele e amigos, como Giovane Silva Ferreira, 13 anos, passam as tardes pescando o peixe, depois levado para uma associação de artesãos **que faz o curtimento da pele do animal**.
	N	Ele e amigos, como Giovane Silva Ferreira, 13 anos, passam as tardes pescando o peixe. Depois, o peixe é levado para uma associação de artesãos. **A associação de artesãos faz o curtimento da pele do animal**.
2	O	Um ser humano, principalmente criança, **que entra em um incêndio sem qualquer treinamento ou proteção**, corre sérios riscos de vida.
	N	Um ser humano **que entra em um incêndio sem qualquer treinamento ou proteção** corre sérios riscos de vida, principalmente se for criança.

6. Second Layer: Rule-based simplification system

This layer is composed of operations to be applied to the text to be simplified in order to make its structure simpler. The operations are applied to each sentence of the text, one sentence at a time. A single sentence may contain more than one linguistic phenomenon at the same time, therefore this sentence should go through the necessary operations in cascade, as described bellow. The output of this system is a XML file containing the simplified sentences and details of operations applied to each sentence.

6.1. Simplification cases

The appropriate operation is applied when any of the 22 linguistic phenomena presented in Table 8 is detected. We use surface information, and morphosyntatic and syntactic clues provided by a parser for Portuguese [Bick 2000] to detect the phenomena to be simplified (these sources of information also assist in the process of simplification). When no phenomenon is detected, the sentence is not simplified.

The operations that can be applied to simplify these phenomena are: (a) split the sentence, (b) change a discourse marker by a simpler and/or more frequent one (the indication is to avoid the ambiguous ones), (c) change passive to active voice, (d) invert the order of the clauses, (e) convert to subject-verb-object ordering, (f) change topicalization and detopicalization of adverbial phrases and (g) non-simplification. We have not implemented some of the operations that were included in the Simplification Annotation Editor (and that consequently are present in our corpus, as described in Section 4): we did not implement strong rewriting because this is not a well-defined operation; dropping sentences or parts of sentences because we aim to apply text summarization techniques ahead of the simplification process; and joining sentences.

Table 8 shows the list of all simplification phenomena covered by our manual, the clues used to identify the phenomena (S = syntactic information; P = punctuation;

lexicalized clues, such as Cj = conjunctions, Pr = relative pronouns, M = discourse markers; Sm = semantic information; NE = named entities), the simplification operations that should be applied in each case, the expected order of clauses in the resulting sentence, and the cue phrases used to replace complex discourse markers or to glue two sentences.

Table 8. The phenomena, clues, operations, order and cue phrases

#	Cases	Clues	Oper.	Order	Cue phrases
1	passive voice clauses	S	c		
2	apposition	S	a	Original/App.	
3	asyndetic coord. clauses	S	a		
4	additives coord. clauses	S,Cj	a		
5	adversative coord. clauses	M	a,b	Main/Coord.	*Mas,*
6	correlated coord. clauses	M	a,b		*também*
7	result coord. clauses	S,M	a,b	Main/Coord.	*Como resultado*
8	explanatory coord. clauses	S,M	a,b		*Isto ocorre porque*
9	causal sub. clauses	M	a,b,d	Sub./Main	*Com isso,*
10	comparative sub. clauses	M	a,b	Main/Sub.	*também*
		M	g		
11	concessive sub. clauses	M	a,b,d	Sub./Main	*Mas*
		M	a,b	Main/Sub.	*Ainda que*
12	conditional sub. clauses	S,M	d	Sub./Main	
13	consecutive sub. clauses	M	a,b	Main/Sub.	*Assim,*
14	final sub. clauses	S,M	a,b	Main/Sub.	*O objetivo é*
15	confirmative sub. clauses	M	a,b,d	Sub./Main	*confirma que*
16	temporal sub. clauses	M	a	Sub./Main	
		M	a,b		*Então,*
17	proportional sub. clauses	M	g		
18	non-finite sub. clauses	S	g		
19	non-rest. relative clauses	S,P,Pr	a	Original/relative	
20	restrictive relative clauses	S,Pr	a	Original/relative	
21	no subject-verb-object order	S	e		
22	adverbial phrases	S,NE,Sm	f	In study	

6.2. Operations

Each sentence of the text is analysed so that the linguistic phenomena are identified and the appropriate operations are called. Each phenomena has a set of simplification operations associated as seen in Table 8.

Splitting the sentence - This operation is the most frequent one. It requires finding the split point in the original sentence (such as the boundaries of relative clauses and appositions, the position of coordinate or subordinate conjunctions) and the creation of a new sentence, whose subject corresponds to the replication of a noun phrase in the original sentence. This operation increases the text length, but decreases the length of the sentences. With the duplication of the term from the original sentence (as subject of the new sentence), the resulting text contains redundant information.

Changing discourse marker - In most cases of subordination and coordination, discourse markers are replaced by most commonly used ones, which are more easily understood.

The selection of discourse markers to be replaced and the choice of new markers (shown in Table 8, col. 4) are done based on the study of [Pardo and Nunes 2006].

Transformation to active voice - Clauses in the passive voice are turned into active voice, with the reordering of the elements in the clause and the modification of the tense and form of the verb. Any other phrases attached to the object of the original sentence have to be carried with it when it moves to the subject position.

Reversing the order of clauses - This operation was primarily designed to handle subordinate clauses, by moving the main clause to the beginning of the sentence, in order to help the reader processing it on their working memory [Graesser et al. 2004]. Each of the subordination cases has a more appropriate order for main and subordinate clauses (as shown in Table 8, col. 3), so that "independent" information is placed before the information that depends on it. This gives the sentence a logical order of the expressed ideas.

Subject-Verb-Object ordering - If a sentence is not in the form of subject-verb-object, it should be rearranged. This operation is based only on information from the syntactic parser. Currently the only case we are treating is the non-canonical order Verb-Object-Subject. We plan to treat other non-canonical orderings in the near future. When performing this operation and the transformation to active voice, a generator of surface forms (GSF) is used to adjust the verb conjugation and regency [Caseli 2007].

Topicalization and detopicalization - This operation is used to topicalize or detopicalize an adverbial phrase. We have not implemented this operation yet, but have observed that moving adverbial phrases to the end or to the front of sentences can make them simpler in some cases. We are still investigating how this operation could be applied, that is, which situations require (de)topicalization.

For an example of application of simplification operations, see example 1 in Table 9. We apply the rule for concessive subordinate clauses (11a) and the sentence is split in two, the clauses are inverted, and a simple discourse marker "Mas" is chosen.

6.3. The cascaded application of operations

As one sentence can present several linguistic phenomena at once, and as the rules were designed to treat a single phenomenon, it is necessary to apply operations in cascade, in order to complete the simplification process for each sentence. After each operation it is necessary to reparse the new sentence and search again for phenomena that need simplification. The standard order of application of the operations is proposed by the Manual of Syntactic Simplification. This order is shown in Figure 2. As an example of a sentence with two cascaded simplification rules consider example 2 in Table 9, where there is passive voice and a non-restrictive relative clause. By the order of application of the operations, we first have simplification S1 and then S2.

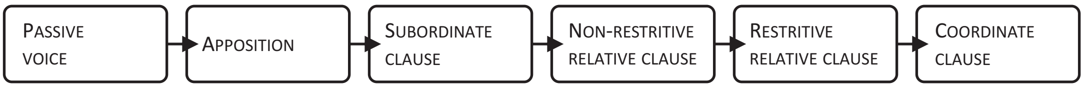

Figure 2. Standard order to recursive process

Table 9. Simplified sentences

#	Case	Type	Example
1	11a	O	O prédio que abriga o Consulado Brasileiro também foi evacuado, embora os diplomatas tenham conseguido permissão para continuar trabalhando.
		S	Os diplomatas conseguiram permissão para continuar trabalhando. Mas o prédio que abriga o Consulado Brasileiro também foi evacuado.
2	1	O	Essa decisão, que tem grande importância para a segurança nacional, deve ser tomada eventualmente por Lula.
		S1	Lula deve tomar eventualmente essa decisão, que tem grande importância para a segurança nacional.
	17	S2	Lula deve tomar eventualmente essa decisão. A decisão tem grande importância para a segurança nacional.

We have so far evaluated the capacity of the rule-based simplifier to identify the phenomena present in each sentence, and to recommend the correct simplification operation. Table 10 presents the performance of the system for this task.

These results are preliminary, since we are still refining our rules. Most of the recall errors on the inversion of clause ordering are due to the absence of a few discourse markers in the list of markers that we use to identify such cases. The majority of recall errors on sentence splitting are due to mistakes on the output of the syntactic parser. The poor performance for subject-verb-object ordering, despite suffering from mistakes of the parser, indicates that our rules for this operation need to be broadened and refined. The same applies to inversion of clause ordering. We did not report performance scores related to the "changing discourse marker" operation because in our evaluation corpus this operation is merged with other types of lexical substitution.

Table 10. Performance on determining the necessary simplification operations

Operation	Precision	Recall	F-measure
Splitting sentences	64.07	82.73	72.17
Inversion of clause ordering	15.40	18.91	16.97
Transformation to active voice	44.29	44.00	44.14
Subject-verb-object ordering	1.12	4.65	1.81
ALL	51.64	65.19	57.62
Non-simplification	64.69	53.58	58.61

In order to assess if the sentences were correctly simplified, it is necessary to do a manual evaluation. It is not possible to automatically compare the output of the rule-based simplifier with the annotated corpus because the sentences in the corpus have passed by operations that are not performed by the simplifier (such as lexical substitution). We are in the process of preparing this manual evaluation phase.

7. Concluding remarks

We have proposed an architecture for text text simplification for Brazilian Portuguese. We distinguish between natural and strong simplification: the first implies the application of as much simplification operations as needed to output a "natural" simplified text, while the second assumes that all operations that can be applied to the sentences will be applied, even if the text may become redundant. This distinction caters for people with different literacy levels. Our architecture handles both types of simplification; it is composed by two layers: the first is a machine-learning system who learns from manually simplified texts when to apply simplification operations to a sentence so that the resulting simplified text is considered natural; the second is a rule-based system that implements all simplification operations and executes them when recommended by the first layer. For strong simplification, the text only needs to pass by the second layer.

We have also presented our first experiments on implementing the first layer. Our classifier for identifying when to split or not a sentence for natural simplification reached 73.5% precision and 73.4% recall on this task using our best performing feature set. In order to refine the natural simplification classification process, we plan to replicate the experiments for other less frequent simplification operations besides sentence splitting.

We have detailed the rules implementing the simplification operations for the linguistic phenomena that we are considering, this is the core of the second layer. We are still refining some of the rules and planning a thorough evaluation of the performance of this layer.

Our text simplification architecture will be the core of two online applications: an authoring tool to help writers to create simplified texts, and a browser plug-in to help low-literacy readers to process online texts.

Acknowledgments

We thank FAPESP and Microsoft Research for supporting the PorSimples project.

References

Aluísio, S. M., Specia, L., Pardo, T. A. S., Maziero, E., Caseli, H. M., and Fortes, R. (2008). A corpus analysis of simple account texts and the proposal of simplification strategies: First steps towards text simplification systems. In *Proceedings of the 26th ACM Symposium on Design of Communication (SIGDOC 2008)*, pages 15–22.

Bick, E. (2000). *The Parsing System "Palavras": Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework.* PhD thesis, Aarhus University.

Caseli, H. M. (2007). *Indução de léxicos bilíngües e regras para a tradução automática.* PhD thesis, Universidade de São Paulo.

Caseli, H. M., Pereira, T. F., Specia, L., Pardo, T. A. S., Gasperin, C., and Aluísio, S. M. (2009). Building a brazilian portuguese parallel corpus of original and simplified texts. *Research in Computing Science. Advances in Computational Linguistics: 10th Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2009)*, 41:59–70.

Chandrasekar, R., Doran, C., and Srinivas, B. (1996). Motivations and methods for text simplification. In *Proceedings of the Sixteenth International Conference on Computational Linguistics (COLING 1996)*, pages 1041–1044.

Chandrasekar, R. and Srinivas, B. (1997). Automatic induction of rules for text simplification. *Knowledge-Based Systems*, 10(3):183–190.

Devlin, S. and Unthank, G. (2006). Helping aphasic people process online information. In *Proceedings of the 8th international ACM SIGACCESS Conference on Computers and Accessibility*, pages 225–226, Portland, USA.

Graesser, A. C., McNamara, D. S., Louwerse, M. M., and Cai, Z. (2004). Coh-Metrix: Analysis of text on cohesion and language. *Behavioral Research Methods, Instruments, and Computers*, 36:193–202.

IBGE (2006). Síntese dos indicadores sociais 2006. http://www.ibge.gov.br/home/estatistica/populacao/condicaodevida/indicadoresminimos/sinteseindicsociais2006/.

INAF (2007). 5 anos - um balanço dos resultados de 2001 a 2005. http://www.acaoeducativa.org.br/base.php?t=pesq_aval_inaf&y=base&z=14.

Inui, K., Fujita, A., Takahashi, T., Iida, R., and Iwakura, T. (2003). Text simplification for reading assistance. In *Proceedings of the Second International Workshop on Paraphrasing: Paraphrase Acquisition and Applications*, pages 9–16.

Klebanov, B. B., Knight, K., and Marcu, D. (2004). Text simplification for information-seeking applications. In *On the Move to Meaningful Internet Systems. Lecture Notes in Computer Science*, volume 3290, pages 735–747. Springer-Verlag.

Max, A. (2006). Writing for language-impaired readers. In *Proceedings of the 7th International Conference on Intelligent Text Processing and Computational Linguistics*, pages 567–570, Mexico City. Springer-Verlag.

Pardo, T. A. S. and Nunes, M. V. (2006). Review and evaluation of DiZer - an automatic discourse analyzer for brazilian portuguese. In *Proceedings of PROPOR 2006. Lecture Notes in Computer Science*, volume 3960, pages 180–189. Springer-Verlag.

Petersen, S. E. and Ostendorf, M. (2007). Text simplification for language learners: A corpus analysis. In *Proceedings of the Speech and Language Technology for Education Workshop (SLaTE-2007)*, pages 69–72, Pennsylvania, USA.

Ramos, W. M. (2006). A compreensão leitora e a ação docente na produção do texto para o ensino a distância. *Linguagem e Ensino*, 9(1):215–242.

SBC (2006). Grandes desafios da pesquisa em computação no brasil: 2006 - 2016. http://www.sbc.org.br/index.php?language=1&content=downloads&id=272.

Siddharthan, A. (2003). *Syntactic Simplification and Text Cohesion*. PhD thesis, University of Cambridge.

Specia, L., Aluísio, S. M., and Pardo, T. A. S. (2008). Manual de simplificação sintática para o português. Technical Report NILC-TR-08-06, NILC.

Vickrey, D. and Koller, D. (2008). Sentence simplification for semantic role labeling. In *Proceedings of the ACL-HLT 2008*, pages 344–352, Columbus, USA.

Williams, S. (2004). *Natural Language Generation of discourse relations for different reading levels*. PhD thesis, University of Aberdeen.

Williams, S. and Reiter, E. (2005). Generating readable texts for readers with low basic skills. In *Proceedings of ENLG 2005*, pages 140–147.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques, 2nd Edition*. Morgan Kaufmann, San Francisco.