# Compressing Multi-document Summaries through Sentence Simplification

Sara Botelho Silveira and António Branco
*University of Lisbon, Lisbon, Portugal*
*Edifício C6, Departamento de Informática, Faculdade de Ciências, Universidade de Lisboa,*
*Campo Grande, 1749-016 Lisboa, Portugal*

Keywords: Multi-document Summarization, Sentence Simplification, Sentence Compression.

Abstract: Multi-document summarization aims at creating a single summary based on the information conveyed by a collection of texts. After the candidate sentences have been identified and ordered, it is time to select which will be included in the summary. In this paper, we propose an approach that uses sentence simplification, both lexical and syntactic, to help improve the compression step in the summarization process. Simplification is performed by removing specific sentential constructions conveying information that can be considered to be less relevant to the general message of the summary. Thus, the rationale is that sentence simplification not only removes expendable information, but also makes room for further relevant data in a summary.

## 1 INTRODUCTION

The increased use of mobile devices brought concerns about text compression, by providing less space for the same amount of text. Compression must be accurate and all the information displayed should be essential. Multi-document text summarization seeks to identify the most relevant information in a collection of texts, complying with a compression rate that determines the length of the summary.

Ensuring at the same time the compression rate and the informativeness of the summary is not an easy task. The most common solution allows the last sentence to be cut in two in the number of words, where the exact compression rate has been reached, compromising the fluency and grammaticality of the summary, and thus the quality of the final text. An alternative is the one where the last candidate sentence is kept in full, surpassing the compression rate. None of these solutions is optimal. Compromising the compression rate by enhancing the quality of the text may not introduce relevant information. Still, compromising the quality of the text can be troublesome for a user wanting to make use of the summary.

Given this, our proposal is to use sentence simplification to compress the extracted sentences down to their main content only, so that more information can fit into the summary, producing a more informative text. After the summarization process has determined the most significant sentences, sentential structures, that are less essential to figure in the summary's short space, can be removed.

The rationale behind using sentence simplification in a summarization context is twofold. On the one hand, it removes expendable information, generating a simpler and easier to read text. On the other hand, it allows the addition of more individual (simplified) sentences to the summary, that otherwise have not been included. Experiments made with human users (Silveira and Branco, 2012b) have shown that simplification indeed helps to improve the summaries produced.

Note that, sentence simplification is also referred in the literature as sentence compression. In this work, the expression ”sentence simplification” is used to define ”sentence compression”, in order to distinguish it from ”compression” itself. We name ”compression” as the step that follows simplification in the summarization process, where the sentences identified as the most relevant ones are selected, based on a predefined compression rate, thus compressing the initial set of sentences contained in the collection of texts submitted as input.

At this point, consider the following list of sentences that can be part of the summary:

Botelho Silveira S. and Branco A..
Compressing Multi-document Summaries through Sentence Simplification.
DOI: 10.5220/0004251201110119
In *Proceedings of the 5th International Conference on Agents and Artificial Intelligence* (ICAART-2013), pages 111-119
ISBN: 978-989-8565-39-6

1. EU leaders signed a new treaty to control budgets on Friday.
2. Only Britain and the Czech Republic opted out of the pact, signed in Brussels at a summit of EU leaders.
3. UK Prime Minister David Cameron, who with the Czechs refused to sign it, said his proposals for cutting red tape and promoting business had been ignored.
4. The countries signed up to a promise to anchor in their constitutions – if possible – rules to stop their public deficits and debt spiralling out of control in the way that led to the eurozone crisis.
5. The treaty must now be ratified by the parliaments of the signatory countries.

This list contains 105 words. However, a compression rate of 80% of the original text states that the summary must only contain 84 words. As the sum of the words of the three first sentences (57 words) does not meet the desired total number of words for the summary, the fourth sentence is also added. Yet, by adding the fourth sentence, the summary makes up 92 words, so the total number of words defined by the compression rate has been surpassed in 9 words. The first option would be to cut the last nine words of the last sentence. That would produce an incorrect sentence.

There are particular constructions that can be removed from these sentences making room for the inclusion of more relevant information. Appositions, parenthetical phrases and relative clauses are examples of those constructions.
Consider, for instance, the following expressions candidates for removal:

- The parenthetical phrase: *signed in Brussels at a summit of EU leaders*
- The relative clause: *who with the Czechs refused to sign*
- The parenthetical phrase: *if possible*

These expressions sum a total of 18 words. The last sentence that has not been added to the summary sums a total of 13 words. So, if all these expressions were removed from the sentences, we would have been able to include in the summary the last sentence. Otherwise that sentence would not be included in the final text, despite being relevant to the overall informativeness of the summary.

The summary, in which sentences have been simplified, contains 84 words and is shown below:

EU leaders signed a new treaty to control budgets on Friday.
Only Britain and the Czech Republic opted out of the pact.
UK Prime Minister David Cameron said his proposals for cutting red tape and promoting business had been ignored.
The countries signed up to a promise to anchor in their constitutions rules to stop their public deficits and debt spiraling out of control in the way that led to the eurozone crisis.
The treaty must now be ratified by the signatory countries' parliaments.

This way, it is possible to produce a comprehensible and fluent summary, that contains the maximum relevant information conveyed by the original collection of texts.

This paper is organized as follows: Section 2 reports the related work; Section 3 overviews the summarization process; Section 4 describes our empirical approach to simplification; Section 5 details the experiments that combine simplification with summarization; and, finally, in Section 6, conclusions are drawn.

## 2 RELATED WORK

Text simplification is an Natural Language Processing (NLP) task that aims at making a text shorter and more readable by simplifying its sentences structurally, while preserving as much as possible the meaning of the original sentence. This task is commonly addressed in two ways: lexical and syntactic simplification. Lexical simplification involves replacing infrequent words by their simpler more common and accessible synonyms. Syntactic simplification, in turn, includes a linguistic analysis of the input texts, that produces detailed tree-structure representations, over which transformations can be made (Feng, 2008).

Previous works (Chandrasekar et al. (1996) and Jing (2000)) have focused on syntactic simplification, targeting specific types of structures identified using rules induced through an annotated aligned corpus of complex and simplified texts.

Jing and McKeown (2000) used simplification in a single-document summarizer, by performing operations, based on the analysis of human abstracts, that remove inessential phrases from the sentences. Blair-Goldensohn et al. (2004) remove appositives and relative clauses in a preprocessing phase of a multi-document summarization process. Another proposal is the one of Conroy et al. (2005), that combine a simplification method, that uses shallow parsing to detect lexical cues that trigger phrase eliminations, with an HMM sentence selection approach, to create multi-document summaries.

Closer to our work is the work of Siddharthan et al. (2004), in which sentence simplification is applied together with summarization. However, they used simplification to improve content selection, that is, before extracting sentences to be summarized. Their simplification system is based on syntactic simplification performed using hand-crafted rules that specify relations between simplified sentences.

Zajic et al. (2007) applied sentence compression techniques to multi-document summarization, using a parse-and-trim approach to generate headlines for news stories. Constituents are removed iteratively from the sentence parse tree, using rules that perform lexical simplification by replacing temporal expressions, preposed adjuncts, determiners, conjunctions, modal verbs -, and syntactic simplification - by selecting specific phenomena in the parse tree.

A different approach was used by Cohn and Lapata (2009), that experimented a tree-to-tree transduction method for sentence compression. They trained a model that uses a synchronous tree substitution grammar, which allows local distortions of a tree topology, used to capture structural mismatches between trees.

A word graph method, to create a single simplified sentence of a cluster of similar or related sentences, was used by Filippova (2010). Considering all the words in these related sentences, a directed word graph is built by linking word *A* to word *B* through an adjacency relation, in order to avoid redundancy. This method was used to avoid redundancy in the summaries produced.

Lloret (2011) proposed a text summarization system that combines textual entailment techniques, to detect and remove information, with term frequency metrics used to identify the main topics in the collection of texts. In addition, a word graphmethod is used to compress and fuse information, in order to produce abstract summaries.

More recently, Wubben et al. (2012) investigated the usage of a machine translation technique to perform sentence simplification. They created a method for simplifying sentences by using Phrase Based Machine Translation, along with a re-ranking heuristic based on dissimilarity. Then, they trained it on a monolingual parallel corpus, and achieved state-of-the-art results. Finally, Yoshikawa et al. (2012) proposed new semantic constraints, to perform sentence compression. These constraints are based on semantic roles, in order to directly capture the relations between a predicate and its arguments.

## 3 SUMMARIZATION PROCESS

The system used is an extractive multi-document summarizer that receives a collection of texts in Portuguese and produces highly informative summaries.

Summarization is performed by means of two main phases executed in sequence: clustering by similarity and clustering by keywords. Aiming to avoid redundancy, sentences are clustered by similarity, and only one sentence from each cluster is selected. Yet, the keyword clustering phase seeks to identify the most relevant content within the input texts. The keywords of the input texts are retrieved and the sentences that are successfully grouped to a keyword cluster are selected to be used in the next step of the process. Furthermore, each sentence has a score, which is computed using the *tf-idf* (term frequency – inverse document frequency) of each sentence word, smoothed by the number of words in the sentence. This score defines the relevance of each sentence and it is thus used to order all the sentences. Afterwards, the simplification process detailed in Section 4 is performed, producing the final summary. A detailed description of this extractive summarization process can be found in (Silveira and Branco, 2012a).

## 4 SIMPLIFICATION PROCESS

In this work, simplification is performed together with compression.

Firstly, from the original input list of sentences, a new list is created, by selecting one sentence at the time, until the total number of words in the list surpasses the maximum number of words determined by the compression rate.

Afterwards, sentences are simplified by removing the expendable information in view of the general summarization purpose. There are a number of structures that can be seen as containing ”elaborative” information about the content already expressed. In this work, five types of structures are targeted:

- Appositions – noun phrases that describe, detail or modify their antecedent (also a noun phrase);
- Adjectives;
- Adverbs or adverb phrases;
- Parentheticals – phrases that explain or qualify other information being expressed;
- Relative clauses – clauses, introduced by a relative pronoun, that modify a noun phrase.

After all these structures have been obtained, by considering the sentence parse tree, they are removed from the original sentence tree. The final step determines if the new simplified sentence can replace the former sentence, based on a specific criteria that takes into account the sentence score.

Due to the fact that simplification removes words from the sentence, once simplification has been performed, new sentences are added to the list of sentences to achieve the maximum number of words of the summary once again. Those newly added sentences are then simplified. This process is repeated while the list is changed or if the compression rate has not been meet.

## 4.1 Structure Identification

In order to perform simplification, a parse tree is created for each sentence, using a constituency parser for Portuguese (Silva et al., 2010). The structures prone to be removed are identified in the tree using Tregex (Levy and Andrew, 2006), a utility for matching patterns in trees. Tregex takes a parse tree and a regular expression pattern. It, then, returns a subtree of the initial tree which top node meets the pattern.

After identifying the subtrees representing each structure, these subtrees are replaced by null trees in the original sentence parse tree, removing its content and generating a new tree without the identified structure. The two steps mentioned above are applied to each of the candidate sentences that have been selected to be a part of the summary.

**Main Clause Selection.** First, the main clause of the sentence is identified. In this phase, other than the next, the desired subtree is selected, ignoring the other subtrees of the main tree. Consider the following sentence and its parse tree:

*No Médio Oriente, apenas Israel saudou a operação.*
In the Middle East, only Israel welcomed the operation.

```
(S
 (S
  (PP (PP (P Em_) (NP (ART o)
    (N′ (N Médio) (N Oriente)))) (PNT ,))
  (S (NP (ADV apenas) (NP (N Israel)))
       (VP (V saudou)
        (NP (ART a) (N operação)))))
 (PNT .))
```

*Apenas Israel saudou a operação.*
Only Israel welcomed the operation.

The expression "*No Médio Oriente*" is ignored, since it is not part of the main clause (in bold). The original sentence is replaced by the simplified one, in which further simplification rules are applied.

The main clause is obtained by applying the following pattern to the parse tree: [S < (VP , NP)]. The main clause S immediately dominates (<) a verb phrase VP that immediately follows (,) a noun phrase NP. This pattern can retrieve several sentences S, contained in a sentence. The main clause selected is the one which is closer to the tree root node. Note that this pattern does not retrieve, for instance, sentences that are not in SVO structure. If the sentence is not in this format, the whole original sentence is used. The main clause is then used to identify the other structures.

**Phrase Compression.** In this stage, the main clause obtained previously is used to identify the structures to be removed. The subtrees of the structures are identified in the sentence parse tree. As was already mentioned, five types of structures are targeted: (i) appositions, (ii) adjectives, (iii) adverbs, (iv) parentheticals, and (v) relative clauses. Examples of every of these structures, and the patterns used are discussed below.

**Appositions** are noun phrases that describe, detail or modify its antecedent (also a noun phrase). Consider the following sentence:

*José Sócrates, primeiro-ministro, e Jaime Gama querem cortar os salários dos seus gabinetes.*
José Sócrates, the Prime Minister, and Jaime Gama want to cut the salaries of their offices.

```
(S (S (S
 (NP (NP (N′ (N José) (N Sócrates))
 (NP
    (PNT ,)
    (NP (N′ (ORD primeiro) (N ministro)))
    (PNT ,)))
  (CONJP (CONJ e) (NP (N′ (N Jaime) (N Gama)))))
 (VP (V querem) (VP (V cortar)
  (PP (P em_) (NP (ART os) (N′ (N salários)
   (PP (P de_) (NP (ART os)
     (N′ (POSS seus) (N gabinetes)))))))))))
(PNT .)))
```

*José Sócrates e Jaime Gama querem cortar os salários dos seus gabinetes.*
José Sócrates and Jaime Gama want to cut the salaries of their offices.

Appositions are identified using two patterns, differing in the punctuation symbol used to enclose the apposition, that can either be a comma or a dash. The comma apposition pattern is: [NP|AP <<, (PNT < ,) <<- (PNT < ,)]. An apposition subtree has a noun phrase NP or (|) an adjective phrase AP as its top node. Its leftmost (<<,) and rightmost (<<-) descendant is a punctuation token PNT that immediately dominates (<) a comma (or a dash).

**Adjectives** qualify nouns or noun phrases, thus being structures prone to be removed. Consider the following sentence:

*O palco tem um pilar central, com 50 metros de altura.*
The stage has a central pillar, 50 meters high.

```
(S
 (NP (ART O) (N palco))
 (VP (V tem)
  (NP (ART um) (N′ (N′ (N pilar) (A central))
  (PP (PNT ,) (PP (P com)
   (NP (CARD 50) (N′ (N metros)
    (PP (P de) (NP (N altura)))))))))
(PNT .))
```

*O palco tem um pilar, com 50 metros de altura.*
The stage has a pillar, 50 meters high.

Adjectives are identified using the pattern: [A $, N]. The subtree top node is an adjective A, which is a sister of a noun N and immediately follows it ($,).

**Adverbs** or adverb phrases are considered differently whether they appear in a noun or in a verb phrase, due to the usage of the adverbs of negation, which precede the verb. The adverbs appearing in a VP are handled differently, to avoid removing negative adverbs and modifying the meaning of the sentence. Consider the following sentence:

*José Sócrates chegou um pouco atrasado ao debate.*
José Sócrates arrived a little late to the debate.

```
(S
 (NP (N′ (N José) (N Sócrates)))
 (VP (VP (V chegou)
   (AP
   (ADVP (ART um) (ADV pouco))
    (A atrasado)))
  (PP (P a_) (NP (ART o) (N debate))))
(PNT .))
```

*José Sócrates chegou atrasado ao debate.*
José Sócrates arrived late to the debate.

The pattern for an adverb occurring in a VP phrase is: [ADV|ADVP , V|VP]. The subtree top node is an adverb ADV or (|) an adverb phrase ADVP, which must follow immediately (,) a verb (V) or a verb phrase (VP). Yet, when the adverb appears in a noun phrase, the pattern is: [ADV|ADVP $ N|NP]. The subtree top node is an adverb ADV or (|) an adverb phrase ADVP that is a sister ($) of a noun N or (|) a noun phrase NP, occurring before or after it.

**Parentheticals** are phrases that explain or qualify other information being expressed. Consider the following sentence:

*O Parlamento aprovou, por ampla maioria, a proposta.*
The Parliament approved by large majority the proposal.

```
(S
 (NP
  (ART O) (N Parlamento))
 (VP
  (V′ (V aprovou)
  (PP
   (PNT ,)
    (PP (P por)
     (NP (N′ (A ampla) (N maioria))))
   (PNT ,)))
  (NP
   (ART a) (N proposta)))
(PNT .))
```

*O Parlamento aprovou a proposta.*
The Parliament approved the proposal.

Parentheticals are identified using several different patterns depending on the punctuation symbol (parenthesis, commas or dashes) that encloses the phrase. The comma patterns is:

[PP|ADV|ADVP|CONJ|CONJP <<, (PNT < ,) <<- (PNT < ,)].

The subtree top node can be either a prepositional phrase PP, or (|) an adverb ADV, or an adverb phrase ADVP, or a conjunction CONJ, or a conjunctional phrase CONJP. Its leftmost descendant (<<,) is a punctuation token PNT that immediately dominates (<) a comma. The same way, its rightmost descendant (<<-) is a punctuation token, immediately dominating a comma.

**Relative clauses** are clauses that modify a noun phrase. They have the same structure as appositions, differing in the top node. Consider the sentence:

*O Parlamento aprovou a proposta, que reduz os vencimentos dos deputados.*
The Parliament approved the proposal, which reduces the salaries of deputies.

```
(S
 (NP (ART O) (N Parlamento))
  (VP
   (V aprovou) (NP (ART a) (N′ (N proposta)
   (CP
    (PNT ,)
     (CP
      (NP (REL que))
       (S
        (VP (V reduz)
        (NP (ART os) (N′ (N vencimentos)
         (PP (P de_) (NP (ART os)
          (N deputados))))))))))))
(PNT .))
```

*O Parlamento aprovou a proposta.*
The Parliament approved the proposal.

The pattern is: `[CP <<, (PNT < ,) <<- (PNT < ,)]`. The subtree top node is a complementizer phrase `CP`, which leftmost (`<<,`) and rightmost (`<<-`) descendants are punctuation tokens `PNT` immediately dominating (`<`) a comma.

### 4.2 Sentence Selection

After the structures have been removed from the sentence, it is time to determine if this new simplified sentence should replace the original one.

Hence, the sentence score is considered. In the summarization algorithm, the sentence score defines the sentence relevance to the complete collection of sentences obtained from the input texts. This score is computed using the *tf-idf* metric, which states that the relevance of a term not only depends on its frequency over the collection of texts, but also it depends on the number of documents in which the term occurs. Equation 1 describes the computation of the sentence score.

$$scores_S = \frac{\sum_w tf-idf_w}{totalWords_S} \qquad (1)$$

Hence, $scores_S$ of the sentence $S$ measures the relevance of this sentence considering the collection of sentences obtained from the input texts.

As words or expressions were removed from the original sentence to create the new simplified sentence, the score of this simplified sentence must be computed, considering only the words that it now contains. After having both sentence scores, the original sentence score is compared with the one of its simplified version. If the simplified sentence score is higher than the one of the original sentence, the simplified sentence replaces the former one in the summary.

This procedure ensures that simplification indeed helps to improve the content of the summary, by including only the simplified sentences that contribute to maximize the informativeness of the final summary.

## 5 EXPERIMENTS

A prototype of the simplification approach detailed above was developed and tested. In this experiment, we used *CSTNews* (Aleixo and Pardo, 2008), an annotated corpus composed by texts in Portuguese. It contains 50 sets of news texts from several domains, for a total of 140 documents, 2,247 sentences, and 47,428 words. Each set contains, on average, three documents that address the same subject. The texts were retrieved from five Brazilian newspapers. In addition, for each set of texts, the corpus contains a manually built summary – the so-called ideal summary – that is used to assess the quality of the automatic summaries. There are 50 ideal summaries, containing a total of 6,859 words. Each ideal summary has on average 137 words, resulting in an average compression rate of 85%.

### 5.1 Simplification Statistics

The simplification process handles several types of syntactic structures that can be removed from a sentence, by executing the simplification algorithm detailed in Section 4. Occurrences of these constructions in the complete corpus were detected and the total number of words they contain was obtained. Statistics of the targeted structures occurring in the complete corpus are shown in Table 1.

Table 1: Corpus targeted structures (totals).

| Phrase type | Total | Total words |
|---|---|---|
| Appositions | 362 | 1,795 |
| Parentheticals | 639 | 3,272 |
| Relative clauses | 178 | 1,446 |
| Adjectives | 999 | 1,199 |
| Adverbs | 288 | 411 |
| Total | 2,466 | 8,123 |

Note that the numbers in this table were obtained by running the simplification process over all the sentences of the corpus. However, there can be more structures in the sentences that are not considered. The reasons are twofold: (i) while retrieving the main clause in each sentence, adjunct expressions may be ignored and (ii) structures can be included in other structures. When obtaining, for instance, a relative clause, inside it can be an adjective or an adverb that will not be taken into account, since the major clause is the one that will indeed be counted.

While retrieving the main clause, 1,200 sentences were found, that include adjunct expressions, containing 12,481 words that can be removed. Considering all the words in the structures – added to the number of words removed when selecting the main clauses –, we retrieve 44% of the words of all texts, a total of 20,694 words. Without considering the main clauses, these structures make up 17% of all texts. Thus, there is a profusion of material to work with.

Table 2 displays the number of structures found, taking into account only the sentences that are considered when executing the simplification algorithm, that is after applying the compression step.

A total of 15,983 words, from the 731 sentences,

Table 2: Simplification structures to be removed.

| Phrase type | Total | Total words |
|---|---|---|
| Appositions | 77 | 400 |
| Parentheticals | 145 | 811 |
| Relative clauses | 26 | 273 |
| Adjectives | 178 | 226 |
| Adverbs | 28 | 39 |
| Total | 454 | 1,749 |

were considered in this simplification process. In this part of the corpora, the first compression step – main clause selection – modifies 197 sentences by removing 3,323 words. This corresponds to an initial compression of 20%, that is, it makes room in the summaries for an additional 158 sentences of average size (21 words).

Additionally, with the removal of the words identified in the structures, the compression can be up to 11%. Adding the compression attained with the main clauses selection (20%) to the one achieved by removing the identified structures (11%), we are able to compress the original texts down to 5,072 words. Considering an average-sized sentence, this number of words corresponds to, on average, 241 sentences of novel information that can be included in the summary.

## 5.2 Comparative Results

For each set of the *CSTNews* corpora, two types of summaries were created using our summarization system: simplified summaries and non-simplified summaries. The simplified summaries are built by including the simplification module in the summarization procedure, and the non-simplified summaries are created without running that module. An example of both a simplified (Figure 1) and a non-simplified summary (Figure 2), for the same set of texts, is presented. As a baseline, GistSumm (Pardo et al., 2003), a single-document summarizer that also performs multi-document summarization, the only available for Portuguese, was used.

In this experiment, we used a compression rate of 85%, since this is the average compression rate of the ideal summaries, which means that the summary contains 15% of the words contained in the set of texts to be summarized.

Afterwards, ROUGE (Lin, 2004) was used to automatically compute precision, recall and F-measure (an average of precision and recall that considers both the same way) metrics. Four ROUGE metrics were computed: ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-S. Although being the most common one, ROUGE-1 cannot be very suitable to the task at stake, since it requires consecutive matches between the summaries being compared. Hence, we also computed all the other three metrics. ROUGE-2 computes 2-gram co-occurrences, allowing some gaps between the sentences. ROUGE-L identifies the common subsequences between two sequences, not requiring consecutive matches. Finally, ROUGE-S computes any pair of words in their sentence order, allowing for arbitrary gaps. Results are shown in Table 3.

Table 3: ROUGE metrics for the automatic summaries.

ROUGE-1

| | Recall | Precision | F-measure |
|---|---|---|---|
| GistSumm | 0.4216 | 0.4776 | 0.4425 |
| Non-Simplified | 0.5400 | 0.5199 | 0.5231 |
| Simplified | 0.5768 | 0.5101 | 0.5357 |

ROUGE-2

| | Recall | Precision | F-measure |
|---|---|---|---|
| GistSumm | 0.2089 | 0.2393 | 0.2204 |
| Non-Simplified | 0.3031 | 0.2953 | 0.2948 |
| Simplified | 0.3535 | 0.3126 | 0.3283 |

ROUGE-L

| | Recall | Precision | F-measure |
|---|---|---|---|
| GistSumm | 0.3832 | 0.4339 | 0.4017 |
| Non-Simplified | 0.5024 | 0.4836 | 0.4866 |
| Simplified | 0.5467 | 0.4832 | 0.5078 |

ROUGE-SU

| | Recall | Precision | F-measure |
|---|---|---|---|
| GistSumm | 0.1760 | 0.2239 | 0.1801 |
| Non-Simplified | 0.2822 | 0.2651 | 0.2594 |
| Simplified | 0.3171 | 0.2499 | 0.2683 |

Some conclusions can be drawn from these results. The complete summarization process has an overall better performance than the baseline, since it overcomes the baseline when considering all the ROUGE values, and all the metrics considered (precision, recall and f-measure). Yet, note that the precision values of the simplified summaries are, in general, lower than the ones of the non-simplified summaries. On the contrary, recall values are, on average, 3 points higher.

On the one hand, the precision values are lower. Intuitively, precision values should decrease, since less in-sequence matches would be found in the simplified summaries. These values are penalized since structures within a sentence are removed, justifying the higher precision values for non-simplified summaries, when compared to the ones of the simplified summaries, concerning most of the metrics in study.

On the other hand, the recall values obtained are very encouraging. These values mean that there is a

> The water level at the river Severn reached 10.4 meters at some point, nearly going over the defense barriers, which are 10.7 meters in height, according to the BBC on Monday, the 23rd.
>
> The rain that's been hitting the United Kingdom has covered roads and millions of people are without electricity and water due to the worst flood the country has seen in the last 60 years.
>
> The worst floods in the last 60 years in the United Kigdom have left thousands of Birtish people homeless.
>
> The United Kingdom's two largest rivers are threatning to overflow this Monday.

Figure 1: Example of a simplified summary (translated from Portuguese) – in its original version, this summary contains 93 words.

> The torrential rain that hit the United Kingdom has covered roads and thousands of people are without electricity and drinking water due to the worst flood the country has seen in 60 years, according to the BBC television network on Monday, the 23rd.
>
> The United Kingdom's largest rivers, the Severn and the Thames, are threatening to overflow this Monday, further aggravating the situation on the central and southern areas of Britain, which have been punished by floods since last Friday.

Figure 2: Example of a non-simplified summary (translated from Portuguese) – in its original version, this summary contains 82 words.

huge density of words that are both in the simplified summaries and in the ideal summaries. So, retrieving the most relevant information in a sentence by discarding the less relevant data ensures that the summary indeed contains the most important conveyed. This is a direct result of the simplification process, since new information is being added to the summary, after sentence simplification has been carried out.

Finally, the f-measure values of the simplified summaries are higher than both the GistSumm ones, and the non-simplified ones, when considering all ROUGE metrics, reflecting the better summaries produced when the simplification process is applied. Note that, ROUGE metrics, other than ROUGE-1, allow for gaps between the sentences, reflecting better results when considering such transformations as the ones performed by the simplification process. The differences between all the precision values are very slightly and do not influence the final f-measure results, which are clearly influenced by the great recall values.

In conclusion, despite removing information from the sentences, the summarization process combined with the simplification approach presented produces highly informative summaries. The combination of these two procedures seeks to create summaries containing not only the most relevant information present in the input texts, but also as much of this information as possible. Moreover, we can conclude that the type of structures that our simplification process removes are indeed additional information to the overall comprehension of the sentence, and their removal makes room in the summary for more novel information.

Thus, our compressing strategy produces high recall summaries. At the same time, the simplified summaries contain the most relevant information, regarding the ideal summaries, achieving our goal of including as much information as possible in each summary.

## 6 FINAL REMARKS

This paper presents an approach that performs syntactic simplification. The rules that make up this simplification approach are fully detailed.

The approach that combines summarization with simplification has proved to be effective, when the complete procedure was evaluated. However, the approach presented modifies the sentences in such a way that currently available automatic metrics are not fair enough when evaluating summaries containing simplified sentences. Thus, a human evaluation is needed to assess the effective gain of simplification.

Even though, recall results are very promising and indicate that simplification allows for the inclusion of further sentences containing relevant information. Despite the precision values are lower than expected, they do not impact on the final f-measure results, which state that the combination between summarization followed by simplification can produce highly informative summaries.

## REFERENCES

Aleixo, P. and Pardo, T. A. S. (2008). CSTNews: Um córpus de textos jornalísticos anotados segundo a teoria discursiva multidocumento CST (cross-document structure theory). Technical report, Universidade de São Paulo.

Blair-Goldensohn, S., Evans, D., Hatzivassiloglou, V., Mckeown, K., Nenkova, A., Passonneau, R., Schiffman, B., Schlaikjer, A., Advaith, Siddharthan, A., and Siegelman, S. (2004). Columbia university at duc 2004. In *Proceedings of the 2004 document understanding conference (DUC 2004)*, HLT/NAACL 2004, pages 23–30, Boston, Massachusetts.

Chandrasekar, R., Doran, C., and Srinivas, B. (1996). Motivations and methods for text simplification. In *In Proceedings of the Sixteenth International Conference on Computational Linguistics (COLING '96)*, pages 1041–1044.

Cohn, T. and Lapata, M. (2009). Sentence compression as tree transduction. *J. Artif. Intell. Res. (JAIR)*, 34:637–674.

Conroy, J., Schlesinger, J., and Stewart, J. (2005). Classy query-based multidocument summarization. In *Proceedings of 2005 Document Understanding Conference*, Vancouver, BC.

Feng, L. (2008). Text simplification: A survey. Technical report, The City University of New York.

Filippova, K. (2010). Multi-sentence compression: finding shortest paths in word graphs. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 322–330, Stroudsburg, PA, USA. ACL.

Jing, H. (2000). Sentence reduction for automatic text summarization. In *Proceedings of the sixth conference on Applied natural language processing*, pages 310–315, Morristown, NJ, USA. Association for Computational Linguistics.

Jing, H. and McKeown, K. R. (2000). Cut and paste based text summarization. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, NAACL 2000, pages 178–185, Stroudsburg, PA, USA. ACL.

Levy, R. and Andrew, G. (2006). Tregex and Tsurgeon: Tools for querying and manipulating tree data structures. In *Proceedings of the 5th Language Resources and Evaluation Conference (LREC)*.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain. ACL.

Lloret, E. (2011). *Text Summarisation based on Human Language Technologies and its Applications*. PhD thesis, Universidad de Alicante.

Pardo, T. A. S., Rino, L. H. M., and das Graças V. Nunes, M. (2003). Gistsumm: A summarization tool based on a new extractive method. In *PROPOR*, Lecture Notes in Computer Science, pages 210–218. Springer.

Siddharthan, A., Nenkova, A., and McKeown, K. (2004). Syntactic simplification for improving content selection in multi-document summarization. In *COLING '04: Proceedings of the 20th international conference on Computational Linguistics*, page 896, Morristown, NJ, USA. ACL.

Silva, J., Branco, A., Castro, S., and Reis, R. (2010). Out-of-the-box robust parsing of Portuguese. In *Proceedings of the 9th Encontro para o Processamento Computacional da Língua Portuguesa Escrita e Falada (PROPOR)*, pages 75–85.

Silveira, S. B. and Branco, A. (2012a). Combining a double clustering approach with sentence simplification to produce highly informative multi-document summaries. In *IRI 2012: 14th International Conference on Artificial Intelligence*, pages 482–489, Las Vegas, USA.

Silveira, S. B. and Branco, A. (2012b). Enhancing multi-document summaries with sentence simplification. In *ICAI 2012: International Conference on Artificial Intelligence*, Las Vegas, USA.

Wubben, S., van den Bosch, A., and Krahmer, E. (2012). Sentence simplification by monolingual machine translation. In *ACL – The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, July 8-14, 2012, Jeju Island, Korea - Volume 1: Long Papers*, pages 1015–1024. The Association for Computer Linguistics.

Yoshikawa, K., Iida, R., Hirao, T., and Okumura, M. (2012). Sentence compression with semantic role constraints. In *ACL – The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, July 8-14, 2012, Jeju Island, Korea - Volume 2: Short Papers*, pages 349–353. The Association for Computer Linguistics.

Zajic, D., Dorr, B. J., Lin, J., and Schwartz, R. (2007). Multi-candidate reduction: Sentence compression as a tool for document summarization tasks. *Inf. Process. Manage.*, 43(6):1549–1570.