A Multilingual Study of Multi-Sentence Compression using Word Vertex-Labeled Graphs and Integer Linear Programming

Elvys Linhares Pontes[1,2,3], Stéphane Huet[2], Juan-Manuel Torres-Moreno[2,3],
Thiago G. da Silva[4,5], Andréa Carneiro Linhares[6]

[1] L3i, University of La Rochelle, La Rochelle, France
[2] LIA, University of Avignon, Avignon, France
[3] Polytechnique Montréal, Montréal, Canada
[4] Inst. Federal de Educação, Ciência e Tecnologia da Paraiba, PB, Brazil
[5] Instituto de Computação Univ. Federal Fluminense, RJ, Brazil
[6] Universidade Federal do Ceará, Sobral, Brazil

elvys.linhares_pontes@univ-lr.fr

Abstract. Multi-Sentence Compression (MSC) aims to generate a short sentence with the key information from a cluster of similar sentences. MSC enables summarization and question-answering systems to generate outputs combining fully formed sentences from one or several documents. This paper describes an Integer Linear Programming method for MSC using a vertex-labeled graph to select different keywords, with the goal of generating more informative sentences while maintaining their grammaticality. Our system is of good quality and outperforms the state of the art for evaluations led on news datasets in three languages: French, Portuguese and Spanish. We led both automatic and manual evaluations to determine the informativeness and the grammaticality of compressions for each dataset. In additional tests, which take advantage of the fact that the length of compressions can be modulated, we still improve ROUGE scores with shorter output sentences.

Keywords. Multi-Sentence Compression, Integer Linear Programming, Word Graph.

1 Introduction

A considerable amount of information is published in various sites every day, e.g. comments, photos, videos and audio in different languages. The increased number of electronic devices (smartphones, tablets, etc.) have made access to these information easier and faster. Moreover, websites such as Wikipedia or news aggregators can provide detailed data on various issues but texts may be long and convey a lot of information. Readers, besides not having the time to go through this amount of information, are not interested in all the proposed subjects and generally select the content of their interest. One solution to this problem is the generation of summaries containing only the key information.

Among the various applications of Natural Language Processing (NLP), Automatic Text Summarization (ATS) aims to automatically identify the relevant data inside one or more documents, and create a condensed text with the main information [21]. At the same time, summaries should be short with as little redundant information as possible. Summarization systems usually rely on statistical, morphological and syntactic analysis approaches [37]. Some of them use Multi-Sentence Compression (MSC) in order to produce from a set of similar sentences a small-sized sentence which is both grammatically correct and informative [1, 10, 21]. Although compression is a challenging task, it is appropriate to generate summaries that are more informative than the state-of-the-art extractive methods for ATS.

The contributions of this article are two-fold. (i) We improved the model for MSC [16] that extends

the common approach based on Graph Theory, using vertex-labeled graphs and Integer Linear Programming (ILP) to select the best compression. The vertex-labeled graphs[1] are used to model a cluster of similar sentences with keywords. (ii) Whereas previous work usually limited the experimental study on one or two datasets, we tested our model on three corpora, each in a different language. Evaluations led with both automatic metrics and human evaluations show that our ILP model consistently generate more informative sentences than two state-of-the-art systems while maintaining their grammaticality. Interestingly, our approach is able to choose the amount of information to keep in the compression output, through the definition of the maximum compression length.

This paper is organized as follows: we describe and survey the MSC problem in Section 2. Next, we detail our approach in Section 3. The experiments and the results are discussed in Sections 4 and 5. Lastly, conclusions and some final comments are set out in Section 6.

2 Related Work

Sentence Compression (SC) aims at producing a reduced grammatically correct sentence. Compressions may have different Compression Ratio (CR) levels,[2] whereby the lower the CR level, the higher the reduction of the information is. SC can be employed in the contexts of the summarization of documents, the generation of article titles or the simplification of complex sentences, using diverse methods such as optimization [7, 8], syntactic analysis, deletion of words [11] or generation of sentences [25, 32]. Recently, many SC approaches using Neural Network (NN) have been developed [25, 32]. These methods may generate good results for a single sentence because they combine many complex structures such as recurrent neural networks (based on Gated Recurrent Units and Long Short Term Memory), the sequence-to-sequence paradigm and condition mechanisms (e.g., attention). However, these composite neural networks need huge corpora to learn how to generate compressions (e.g., Rush et al. used the Gigaword corpus that contains around 9.5 million news) and take a lot of time to accomplish the learning process.

Multi-Sentence Compression (MSC), also coined as Multi-Sentence Fusion, is a variation of SC. Unlike SC, MSC combines the information of a cluster of similar sentences to generate a new sentence, hopefully grammatically correct, which compresses the most relevant data of this cluster. The idea of MSC was introduced by Barzilay and McKeown [3], who developed a multi-document summarizer which represents each sentence as a dependency tree; their approach aligns and combines these trees to fusion sentences. Filippova and Strube [12] also used dependency trees to align each cluster of related sentences and generated a new tree, this time with ILP, to compress the information. In 2010, Filippova presented a new model for MSC, simple but effective, which is based on Graph Theory and a list of stopwords. She used a Word Graph (WG) to represent and to compress a cluster of related sentences; the details of this model, which is extended by the work of this paper, can be found in Section 2.1.

Inspired by the good results of the Filippova's method, many studies have used it in a first step to generate a list of the N shortest paths, then have relied on different reranking strategies to analyze the candidates and select the best compression [1, 5, 23, 38]. Boudin and Morin [5] developed a reranking method measuring the relevance of a candidate compression using *key phrases*[3], obtained with the TextRank algorithm [26], and the length of the sentence. Another reranking strategy was proposed by Luong et al. [23]. Their method ranks the sentences from the counts of unigrams[4] occurring in every source sentence. ShafieiBavani et al. [34] also used a WG model; their approach consists of three main components: (i) a merging

[1] A vertex-labeled graph means a graph where each node has a label. In this work, a label is represented by a color and different nodes can have the same label.

[2] The CR is the length of the compression divided by the average length of all source sentences

[3] *key phrases* are words that capture the main topics of a document.

[4] An n-gram is a contiguous sequence of n items from a given text.

stage based on Multiword Expressions (MWE), (ii) a mapping strategy based on synonymy between words and (iii) a reranking step to identify the best compression candidates generated using a Part-of-Speech-based language model (POS-LM). Tzouridis et al. [38] proposed a structured learning-based approach. Instead of applying heuristics as Filippova [10], they adapted the decoding process to the data by parameterizing a shortest path algorithm. They devised a structural support vector machine to learn the shortest path in possibly high dimensional joint feature spaces and proposed a generalized loss-augmented decoding algorithm that is solved exactly by ILP in polynomial time.

Linhares Pontes et al. [16] also presented an ILP approach that models a set of similar sentences as vertex-labeled word graphs. Their approach selects keywords and relevant 3-grams to generate more informative compressions while maintaining their grammaticality as possible. They have studied the quality of compressions by analyzing different amounts of keywords in order to manage both the length and the informativeness of compressions.

We found two other studies that applied ILP to combine and compress several sentences. Banerjee et al. [1] developed a multi-document ATS system that generated summaries after compressing similar sentences. They used Filippova's method to generate 200 random compressed sentences. Then they created an ILP model to select the most informative and grammatically correct compression. Thadani and McKeown [36] proposed another ILP model using an inference approach for sentence fusion. Their ILP formulation relies on n-gram factorization and aims at avoiding cycles and disconnected structures.

In the ATS task, Shang et al. [35] adapted the Boudin and Morin's approach [5] to take into account the grammaticality for the reranking of compressions. Instead of the TextRank algorithm, they analyze the spreading influence in WG to generate more informative and grammatical compressions and to improve the quality of summaries. Nayeem et al. [28] designed a paraphrastic sentence fusion model which jointly performs sentence fusion and paraphrasing using skip-gram word embedding model at the sentence level.

Recently, Zhao et al. [39] presented an unsupervised rewriter to improve the grammaticality of MSC outputs while introducing new words. They used the WG approach to produce coarse-grained compressions, from which they substitute words with their shorter synonyms to yield paraphrased sentence. Then, their neural rewriter proposes paraphrases for these compressions in order to improve grammaticality and encourage more novel words.

Another related task is the sentence aggregation that combines a group of sentences, not necessarily with a similar semantic content, to generate a single sentence (e.g., "*The car is here.*" and "*It is blue.*" can be aggregated into "*The blue car is here.*"). This aggregation can be at semantic and syntactic levels [31]. The aggregation rules can be acquired automatically from a corpus [2]. However, this process is not possible for all situations and the sentence aggregation depends on the sentence planning to combine the sentences.

Following previous studies for MSC that rely on Graph Theory with good results, this work presents a new ILP framework that takes into account keywords for MSC. We compare our learning approach to the graph-based sentence compression techniques proposed by Filippova [10] and Boudin and Morin [5], considered as state-of-the-art methods for MSC. We intend to apply our method on various languages and not to be dependent on linguistic resources or tools specific to languages. This led us to put aside systems which, despite being competitive, rely on resources like WordNet or Multiword expression detectors [34]. Since we borrowed concepts and ideas from Filippova's method, we detail her approach in the next section.

2.1 Filippova's Method

Filippova [10] modeled a document D containing n similar sentences $\{s_1, s_2, \ldots, s_n\}$, as a directed word graph $G = (V, A)$. V is the set of vertices (words) and A is the set of arcs (adjacency relationship). Figure 1 illustrates the word graph of the following Portuguese sentences:

1. *George Solitário, a última tartaruga gigante Pinta Island do mundo, faleceu.* (Lonesome George, the world's last Pinta Island giant tortoise, has passed away.)

2. *A tartaruga gigante conhecida como George Solitário morreu domingo no Parque Nacional de Galapagos, Equador.* (The giant tortoise known as Lonesome George died Sunday at the Galapagos National Park in Ecuador.)

3. *Ele tinha apenas cem anos de vida, mas a última tartaruga gigante Pinta conhecida, George Solitário, faleceu.* (He was only about a hundred years old, but the last known giant Pinta tortoise, Lonesome George, has passed away.)

4. *George Solitário, a última tartaruga gigante da sua espécie, morreu.* (Lonesome George, a giant tortoise believed to be the last of his kind, has died.)

The initial graph G is composed of the first sentence (1) and the vertices –begin– and –end–. For a new sentence, a new vertex is created when a word/POS pair cannot be matched to an existing vertex of G once lowercased. Besides, at most one occurrence of a given word/POS inside a sentence can be associated with a given vertex.

Sentences are individually analyzed and added to G. Each sentence represents a simple path between the –begin– and –end– vertices and its words are inserted in the following order:

1. Non-stopwords for which no candidate exists in the graph or for which an unambiguous mapping is possible;

2. Non-stopwords for which there are several possible candidates in the graph that may occur more than once in the sentence;

3. Stopwords.

In cases 2 and 3, the word mapping is ambiguous because there is more than one vertex in the graph that references the same word/POS. In this case, we analyze the immediate context (the preceding and following words/POSs in the sentence and the neighboring nodes in the graph) or the frequency (i.e., the number of words that were mapped to the considered vertex) to select the best candidate node.

Once vertices have been added, arcs are valued by weights which represent the levels of cohesion between two words in the graph (Equation 1). Cohesion is calculated from the frequency and the position of these words in sentences, according to Equation 2:

$$w(i,j) = \frac{\text{cohesion}(i,j)}{\text{freq}(i) \times \text{freq}(j)}, \quad (1)$$

$$\text{cohesion}(i,j) = \frac{\text{freq}(i) + \text{freq}(j)}{\sum_{s \in D} \text{diff}(s,i,j)^{-1}}, \quad (2)$$

where $\text{freq}(i)$ is the word frequency mapped to the vertex i and the function $\text{diff}(s,i,j)$ refers to the distance between the offset positions of words i and j in the sentences s of D containing these two words.

From the graph G, the system calculates the 50 shortest paths that are longer than eight words and have at least one verb. Finally, the system reranks the paths by normalizing the total path weight over their length and selects the path with the lowest score as the best MSC.

3 Our Approach

Filippova's method chooses the path with the lowest score taking into account the level of cohesion between two adjacent words in the document. However, two words with a strong cohesion do not necessarily have a good informativeness because the cohesion only measures the distance and the frequency of words in the sentences. In this work, we propose a method to concurrently analyze cohesion and keywords in order to generate a more informative and comprehensible compression.

Our method calculates the shortest path from the cohesion of words and grants bonuses to the paths that have different keywords. For this purpose, our approach is based on Filippova's method (Section 2.1) to model a document D as a graph and to calculate the cohesion of words. In addition, we analyze the keywords of the document to favor hypotheses with meaningful information.

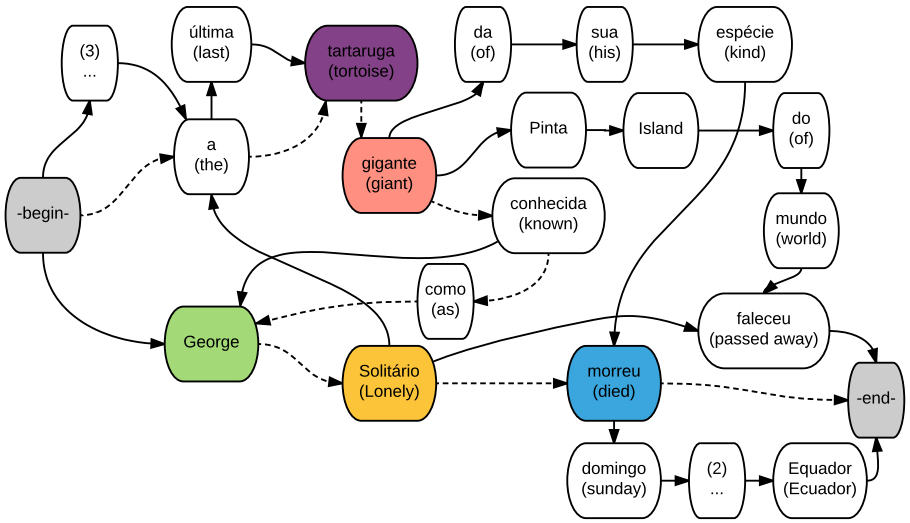

Fig. 1. WG generated from the sentences (1) to (4) (without the punctuation and Part-of-Speech (POS) for easy readability). The dotted path represents the best compression for this WG and the colored vertices represent the keywords of the document.

3.1 Keyword Extraction

Introducing keywords in the graph helps the system to generate more informative compressions because it takes into account the words that are representative of the cluster to calculate the best path in the graph, and not only the cohesion and frequency of words. Keywords can be identified for each cluster with various extraction methods and we study three widely used techniques: Latent Semantic Indexing (LSI), Latent Dirichlet Allocation (LDA) and TextRank. Despite the small number of sentences per cluster, these methods generate good results because clusters are composed of similar sentences with a high level of redundancy. LSI uses Singular-Value Decomposition (SVD), a technique closely related to eigenvector decomposition and factor analysis, to model the associative relationships [9]. LDA is a topic model that generates topics based on word frequency from a set of documents [4]. Finally, TextRank algorithm analyzes the words in texts using WGs and estimates their relevance [26]. For LDA whose modeling is based on the concept of topics, we consider that the document D describes only one topic since it is composed of semantically close sentences related to a specific news item. A same word or keyword can be represented by one or several nodes in WGs (see Section 2.1). In order to prioritize the sentence generation containing multiple keywords and to reduce the redundancy, we add a bonus to the compression score when the compression contains different keywords.

3.2 Vertex-Labeled Graph

A vertex-labeled graph is a graph $G = (V, A)$ with a label on the vertices $K = \{0, ..., |K|\}$, where $|K|$ is the number of different labels. This graph type has been employed in several domains such as biology [40] or NLP [6]. In this last study, the correction of Wikipedia inter-language links was modeled as a Colorful Components problem. Given a vertex-colored graph, the Colorful Components problem aims at finding the minimum-size edge sets that are connected and do not have two vertices with the same color.

In the context of MSC, we want to generate a short informative compression where keyword may be represented by several nodes in the word graph. Labels enable us to represent keywords in vertex-labeled graphs and generate a compression without repeated keywords while preserving the informativeness. In this framework, we grant

bonuses only once for nodes with the same label to prioritize new information in the compression (Figure 1). To make our model coherent, we added a base label (label 0) for all non-keywords in the word graph. The following section describes our ILP model to select sentences including labeled keywords inside WGs.

3.3 ILP Modeling

There are several algorithms with a polynomial complexity to find the shortest path in a graph. However, the restriction on the minimum number $P_{\min}$ of vertices (i.e., the minimum number of words in the compression) makes the problem NP-hard. Indeed, let v_0 be the –begin– vertex. If $P_{\min}$ equals $|V|$ and if we add an auxiliary arc from –end– vertex to v_0, our problem is similar to the Traveling Salesman Problem (TSP), which is NP-hard.

For this work we use the formulation known as Miller-Tucker-Zemlin (MTZ) to solve our problem [30, 36]. This formulation uses a set of auxiliary variables, one for each vertex in order to prevent a vertex from being visited more than once in the cycle and a set of arc restrictions.

The problem of production of a compression that favors informativeness and grammaticality is expressed as Equation 3. In other words, we look for a path (sentence) that has a good cohesion and contains a maximum of labels (keywords).

$$\text{Minimize} \Big(\sum_{(i,j)\in A} w(i,j) \cdot x_{i,j} - c \cdot \sum_{k \in K} b_k \Big) \quad (3)$$

where x_{ij} indicates the existence of the arc (i,j) in the solution, $w(i,j)$ is the cohesion of the words i and j (Equation 1), K is the set of labels (each representing a keyword), b_k indicates the existence of a word with label (keyword) k in the solution and c is the keyword bonus of the graph.[5]

[5]The keyword bonus allows the generation of longer compressions that may be more informative.

3.4 Structural Constraints

We describe the structural constraints for the problem of consistency in compressions and define the bounds of the variables. First, we consider the problem of consistency which requires an inner and an outer arc active for every word used in the solution, where y_v indicates the existence of the vertex v in the solution.

$$\sum_{i \in \delta^+(v)} x_{vi} = y_v \qquad \forall v \in V, \quad (4)$$

$$\sum_{i \in \delta^-(v)} x_{iv} = y_v \qquad \forall v \in V. \quad (5)$$

The constraints (6) and (7) control the minimum and the maximum number of vertices ($P_{\min}$ and $P_{\max}$) used in the solution respectively, i.e., the minimum and the maximum number of words in the final compression.

$$\sum_{v \in V} y_v \geq P_{\min}, \quad (6)$$

$$\sum_{v \in V} y_v \leq P_{\max}. \quad (7)$$

The set of constraints (8) matches label variables (keywords) with vertices (words), where $V(k)$ is the set of all vertices with label k.

$$\sum_{v \in V(k)} y_v \geq b_k, \qquad \forall k \in K. \quad (8)$$

Equality (9) sets the vertex v_0 in the solution.

$$y_0 = 1. \quad (9)$$

The restrictions (10) and (11) are responsible for the elimination of sub-cycles, where u_v ($\forall v \in V$) are auxiliary variables for the elimination of sub-cycles and M is a large number (e.g., $M = |V|$).

$$u_0 = 1, \tag{10}$$

$$u_i - u_j + 1 \leq M - M \cdot x_{ij} \quad \forall (i,j) \in A, j \neq 0. \tag{11}$$

Finally, equations (12) – (14) define the field of variables.

$$x_{ij} \in \{0,1\}, \qquad \forall (i,j) \in A, \tag{12}$$

$$y_v \in \{0,1\}, \qquad \forall v \in V, \tag{13}$$

$$u_v \in \{1,2,\ldots,|V|\}, \qquad \forall v \in V. \tag{14}$$

We calculate the 50 best solutions according to the objective (3) having at least eight words and at least one verb. Specifically, we find the best solution, then we add a constraint in the model to avoid this solution and repeat this process 50 times to find the other solutions.

The optimized score (Equation 3) explicitly takes into account the size of the generated sentence. Contrary to Filippova's method, sentences may have a negative score because we subtract from the cohesion value of the path the introduced scores for keywords. Therefore, we use the exponential function to ensure a score greater than zero. Finally, we select the sentence with the lowest final score (Equation 15) as the best compression.

$$\text{score}_{norm}(s) = \frac{e^{\text{score}_{opt}(s)}}{||s||}, \tag{15}$$

where $\text{score}_{opt}(s)$ is the score of the sentence s from Equation 3.

4 Experimental Setup

Algorithms were implemented using the Python programming language with the takahe[6] and gensim[7] libraries. The mathematical model was implemented in C++ with the Concert library and we used the solver CPLEX 12.6[8].

[6] http://www.florianboudin.org/publications.html

[7] https://radimrehurek.com/gensim/models/ldamodel.html

[8] https://www.ibm.com/products/ilog-cplex-optimization-studio

The objective function (see Equation 3) involves a keyword bonus. Since each WG can have weight arcs of different values, fixing this bonus is decisive to allow the generation of slightly longer compressions. We tested several metrics (fixed values, the arithmetic average, the median, and the geometric average of the weights arcs of WG) to define the keyword bonus of the WG and empirically found that geometric mean outperformed others.

4.1 Evaluation Datasets

Various corpora have been developed for MSC and are composed of clusters of similar sentences from different source news in English, French, Portuguese, Spanish or Vietnamese languages. Whereas the data built by McKeown et al. [24] and Luong et al. [23] have clusters limited to pairs of sentences, the corpora made by Filippova [10], Boudin and Morin [5], and Linhares Pontes et al. [22] contain clusters of at least 7 similar sentences. McKeown et al. [24] collected 300 English sentence pairs taken from newswire clusters using Amazon's Mechanical Turk, while the corpus introduced in Luong et al. [23] is made of 250 Vietnamese sentences divided into 115 groups of similar sentences with 2 sentences by group. McKeown et al. [24], Luong et al. [23], Boudin and Morin [5], and Linhares Pontes et al. [22] made their corpora publicly available, but only the data associated with these last two articles are more suited to the multi-document summarization or question-answering tasks because the documents to analyze are usually composed of many similar sentences. Therefore, we use these two corpora made of French, Portuguese and Spanish sentences.

Table 1 summarizes the statistics of this set of data having 40 clusters of sentences for each language. The Type-Token Ratio (TTR) indicates the reuse of tokens in a cluster and is defined by the number of unique tokens divided by the number of tokens in each cluster; the lower the TTR, the greater the reuse of tokens in the cluster. The

sentence similarity represents the average cosine similarity of the sentences in a cluster.[9]

The French corpus has 3 sentences compressed by native speakers for each cluster, references having a compression rate (CR) of 60%. Like the French corpus, the Portuguese and Spanish corpora are composed of the first sentences of the articles found in Google News. Each cluster is composed of related sentences and was chosen among the first sentence from different articles about Science, Sport, Economy, Health, Business, Technology, Accidents/Catastrophes, General Information and other subjects. A cluster has at least 10 similar sentences by topic and 2 reference compressions made by different native speakers. The average CRs are 54% and 61% for the Portuguese and the Spanish corpora, respectively.

The three languages derive from Latin and are closely related languages. However, they differ in many details of their grammar and lexicon. Moreover, the datasets produced for the three languages are unlike according to several features. First, the corpus made by Linhares Pontes et al. [22] contains a smaller (Portuguese corpus) and a larger (Spanish corpus) dataset in terms of sentences than the French corpus. Besides, the compression rates of the three datasets indicate that the Portuguese source sentences have more irrelevant tokens. The sentence similarity (Table 1, second last line) describes the variability of sentences in the source sentences and in the references, and reflects here that the sentences are slightly more diverse for the French corpus. This translates into a higher TTR observed for the French part (38.8%) than for the two other languages (33.7% and 35.2%).

4.2 Automatic and Manual Evaluations

The most important features of MSC are informativeness and grammaticality. Informativeness measures how informational is the generated text. As references are assumed to contain the key information, we calculated informativeness scores counting the n-grams in common between the system output and the reference compressions using ROUGE [14]. In particular, we used the metrics ROUGE-1 and ROUGE-2, F-measure being preferred to recall for a fair comparison of various lengths of compressed sentences. Like in [5], ROUGE metrics are calculated with stopwords removal and stemming[10].

Due to limitations of the ROUGE systems that only analyze unigrams and bigrams, we also led a manual evaluation with four native speakers for French, Portuguese and Spanish. The native speakers of each language evaluated the compression in two aspects: informativeness and grammaticality. In the same way as Filippova [10] as well as Boudin and Morin [5], the native speakers evaluated the grammaticality in a 3-point scale: 2 points for a correct sentence; 1 point if the sentence has minor mistakes; 0 point if it is none of the above. Like grammaticality, informativeness is evaluated in the same range: 2 points if the compression contains the main information; 1 point if the compression misses some relevant information; 0 point if the compression is not related to the main topic.

5 Experimental Assessment

Compression rates are strongly correlated with human judgments of meaning and grammaticality [27]. On the one hand, too short compressions may compromise sentence structure, reducing the informativeness and grammaticality. On the other hand, longer compressions may be more interesting for ATS when informativeness and grammaticality are decisive features. Consequently, we analyze compression with multiple maximum compression lengths (50%, 60%, 70%, 80%, 90% and ∞, the last value meaning that no constraint is fixed on the output size).

Following the idea proposed by ShafieiBavani et al. [34] and already implemented with success in other domains such as speech recognition (e.g., [13]), we tested the use of a POS-based Language Model (POS-LM) as a post-processing stage in order to improve the grammaticality of

[9] The cosine similarity between two vectors u and v associated with two sentences is defined by $\frac{u \cdot v}{||u|| \, ||v||}$ in the [0,1] range.

[10] http://snowball.tartarus.org/

Table 1. Statistics of the corpora.

Characteristics	French		Portuguese		Spanish	
	Source	References	Source	References	Source	References
#tokens	20,224	2,362	17,998	1,425	30,588	3,694
#vocabulary (tokens)	2,867	636	2,438	533	4,390	881
#sentences	618	120	544	80	800	160
avg. sentence length (tokens)	33.0	19.7	33.1	17.8	38.2	23.1
type-token ratio (TTR)	39%	50%	34%	68%	35%	43%
sentence similarity	0.46	0.67	0.51	0.59	0.47	0.64
compression rate	—	60%	—	54%	—	61%

compressions. Specifically, for each cluster, the ten best compressions according to our optimized score are reranked by a 7-gram POS-LM trained with the SRILM toolkit[11] on the French, Portuguese and Spanish parts of the Europarl dataset,[12] tagged with TreeTagger [33].

5.1 Results

Since our method strongly depends on the set of keywords to generate informative compressions, we investigate the performance of the three keyword methods (LDA, LSI and TextRank), selecting the 5 or 10 most relevant words. We verified the percentage of keywords generated by these methods that are included in the reference compression (Table 2). A significantly higher rate of keywords in the references is observed when using LDA or LSI instead of TextRank. In order to obtain the most relevant words in a cluster with different sizes, we used LDA in our final MSC system to identify 10 keywords for each cluster.

Tables 3, 4 and 5 describe the ROUGE recall scores measured for Filippova's [10] method (named F10), Boudin and Morin's [5] method (named BM13) and our method with multiple maximum compression lengths. As for each CR setup the size of the outputs to evaluate are comparable, the recall scores are preferred in this case to measure the information retained in compressions. First, let us note that CRs effectively observed may differ from the fixed value of P_{max}. For example, a 50% threshold leads to real CRs of 38% to 40% for all languages, while an 80% level creates new sentences with real CRs between 53% and 60%. Interestingly, our system obtained better ROUGE recall scores than both baselines in all languages for comparable compression lengths. If we prioritize meaning, our method with no explicit constraint on the maximum compression length (ILP:∞) improved the compression quality with a small increase of the compression length (compression ratio between 55.4% and 65.9%). Instead, we can limit the length and generate compressions that are shorter and have still better ROUGE scores than the baselines.

Based on these results, a further analysis was done for the 80% and ∞ configurations.

Table 2. Percentage of keywords included in the reference compression for French, Portuguese and Spanish corpora.

Methods	fr	pt	es
LDA: 5 kws	**91%**	**88%**	**85%**
LSI: 5 kws	90%	87%	81%
TextRank: 5 kws	69%	55%	58%
LDA: 10 kws	**84%**	**70%**	**76%**
LSI: 10 kws	**84%**	69%	73%
TextRank: 10 kws	56%	44%	50%

[11] http://www.speech.sri.com/projects/srilm/

[12] http://www.statmt.org/europarl/

Table 3. ROUGE recall scores for multiple maximum compression lengths using the French corpus.

	French		
Methods	**ROUGE-1**	**ROUGE-2**	**CR**
F10	0.5971	0.4072	51.3%
BM13	0.6740	0.4695	59.8%
ILP:50%	0.4763	0.3039	39.1%
ILP:60%	0.5990	0.4101	47.4%
ILP:70%	0.6420	0.4206	53.5%
ILP:80%	0.6783	0.4573	60.0%
ILP:90%	0.6981	**0.4758**	61.8%
ILP:∞	**0.7010**	0.4751	62.6%

Table 4. ROUGE recall scores for multiple maximum compression lengths using the Portuguese corpus.

	Portuguese		
Methods	**ROUGE-1**	**ROUGE-2**	**CR**
F10	0.5354	0.2935	52.2%
BM13	0.6304	0.3493	69.1%
ILP:50%	0.4689	0.2521	40.0%
ILP:60%	0.5369	0.2967	48.1%
ILP:70%	0.5652	0.3088	54.0%
ILP:80%	0.6056	0.3321	59.0%
ILP:90%	0.6341	0.3492	64.6%
ILP:∞	**0.6407**	**0.3546**	65.9%

Table 6[13] describes the results for the French, Portuguese and Spanish corpora using ROUGE F-measure scores. The first two columns display the evaluation of the two baseline systems; the ROUGE scores measured with our method using either 80% or ∞ maximum compression lengths are shown in the next two columns and the last two columns respectively. The outputs produced by all of these systems for two sample clusters in Spanish and Portuguese

[13]Although we used the same system and data as Boudin and Morin [5] for the French corpus, we were not able to exactly reproduce their results. The ROUGE F-measure scores given in their article are close to ours for their system: 0.6568 (ROUGE-1), 0.4414 (ROUGE-2) and 0.4344 (ROUGE-SU4), but using F10 we measured higher scores than them: 0.5744 (ROUGE-1), 0.3921 (ROUGE-2) and 0.3700 (ROUGE-SU4).

Table 5. ROUGE recall scores for multiple maximum compression lengths using the Spanish corpus.

	Spanish		
Methods	**ROUGE-1**	**ROUGE-2**	**CR**
F10	0.4437	0.2631	43.2%
BM13	0.5167	0.2981	61.2%
ILP:50%	0.3814	0.1990	38.7%
ILP:60%	0.4594	0.2651	45.3%
ILP:70%	0.5050	0.2922	50.2%
ILP:80%	0.5191	0.2982	53.2%
ILP:90%	0.5242	0.2982	54.4%
ILP:∞	**0.5305**	**0.3036**	55.4%

can be found in the Appendix. Globally, all versions of our ILP method outperform both baselines according to ROUGE F-measures for the Portuguese and Spanish corpora, and our ILP systems (ILP:80% and ILP:∞) obtained similar results to BM13 for the French corpus. The POS-LM post-processing further improved the ROUGE scores for Portuguese and Spanish, but unfortunately not for the French corpus.

Table 7 displays the average length, the compression ratio and the average number of keywords that are kept in the final compression. F10 generated the shortest compressions for all corpora, our approach producing outputs of an intermediate length with respect to BM13, except for the French corpus for which ILP:∞ generated slightly longer compressions. The keyword bonus and the POS-LM score act differently on the selection of words. On the one hand, the keyword bonus promotes the integration of keywords from difference sentences. On the other hand, the POS-LM favors grammaticality and longer subsequences of the original sentences, which reduces the mix of sentences and, consequently, the number of keywords in the compressions.

We also led a manual evaluation to study the informativeness and grammaticality of compressions. We measured the inter-rater agreement on the judgments we collected, obtaining values of Fleiss' kappa of 0.423, 0.289 and 0.344 for French, Portuguese and Spanish respectively. These results show that human evaluation is rather subjective. Questioning evaluators on how they

Table 6. ROUGE F-measure results on the French, Portuguese and Spanish corpora. The best ROUGE results are in bold.

Metrics	Methods					
	F10	BM13	ILP:80%	ILP:80%+LM	ILP:∞	ILP:∞+LM
French						
ROUGE-1	0.6384	0.6674	0.6630	0.6418	**0.6730**	0.6460
ROUGE-2	0.4423	**0.4672**	0.4487	0.4187	0.4567	0.4179
ROUGE-SU4	0.4297	**0.4602**	0.4410	0.4152	0.4511	0.4136
Portuguese						
ROUGE-1	0.5388	0.5532	0.5668	0.5763	0.5700	**0.5811**
ROUGE-2	0.2971	0.3029	0.3105	0.3112	0.3132	**0.3249**
ROUGE-SU4	0.2938	0.2868	0.3060	0.3149	0.3057	**0.3210**
Spanish						
ROUGE-1	0.5004	0.5140	0.5422	**0.5500**	0.5425	0.5442
ROUGE-2	0.2983	0.2960	0.3128	**0.3195**	0.3109	0.3194
ROUGE-SU4	0.2847	0.2801	0.2973	**0.3052**	0.2963	0.3047

proceed to rate sentences reveals that they often made their choice by comparing outputs for a given cluster.

Table 8 shows the manual analysis that ratifies the good results of our system. Informativeness scores are consistently improved by the ILP method, whereas grammaticality results measured on the three systems are similar. Besides, statistical tests show that this enhancement regarding informativeness and grammaticality is significant for Spanish corpus. For the Portuguese and Spanish corpora, our method obtained the best results for informativeness and grammaticality with shorter compressions. For the French corpus, F10 obtained the highest value for grammatical quality, while BM13 generated more informative compressions. Finally, the reranking method proposed by BM13 based on the analysis of *key phrases* of candidate compression improves informativeness, but not to the same degree as our ILP model. This more moderate enhancement can be related to the limitation of this reranking method to candidate sentences generated by F10.

5.2 Discussion

Short compressed sentences are appropriate to summarize documents; however, they may remove key information and prejudice the informativeness of the compression. For instance, for the sentences that would be associated with a higher relevant score by the ATS system, producing longer sentences would be more appropriate. Generating longer sentences makes easier to keep informativeness but often increases difficulties to have a good grammatical quality while combining different parts of sentences. Depending on the kind of cluster short compressions can be generated or not with good informativeness scores. In that respect, the system has to adapt its analysis to generate long or short sentences.

F10 produced the shortest compressions for all corpora but its outputs have the worst informativeness score. BM13 improved these results; however, their compressions are longer than F10 (for all corpora) and our system (for the Portuguese and the Spanish corpora). For Spanish, the informativeness scores of all versions of our method are statistically better than F10, and the version ILP:∞+LM is statistically better than both baselines for this corpus. Given the small difference of informativeness between BM13 and our ILP approach for the Portuguese and the French corpora, we analyzed the relation between informativeness and CR to define which method

Table 7. Compression length (#words), standard deviation and number of used keywords computed on the French, Portuguese and Spanish corpora.

Metrics	Methods					
	F10	BM13	ILP:80%	ILP:80%+LM	ILP:∞	ILP:∞+LM
French						
Avg. Length	16.9 ± 5.1	19.7 ± 6.9	19.8 ± 4.8	19.5 ± 4.9	20.6 ± 5.5	20.8 ± 5.8
Comp. Ratio. (%)	51.3	59.8	59.9	59.2	62.6	63.1
Keywords	6.8	7.7	8.3	7.9	8.5	8.1
Portuguese						
Avg. Length	17.3 ± 5.3	22.9 ± 6.3	19.5 ± 4.0	19.4 ± 4.4	21.8 ± 5.5	20.5 ± 5.0
Comp. Ratio. (%)	52.2	69.1	59.0	58.7	65.9	62.2
Keywords	7.0	8.5	8.2	8.0	8.9	8.3
Spanish						
Avg. Length	16.5 ± 6.4	23.4 ± 8.4	20.3 ± 5.9	20.9 ± 5.2	21.1 ± 7.0	23.4 ± 7.3
Comp. Ratio. (%)	43.2	61.2	53.2	54.7	55.4	61.2
Keywords	5.8	6.9	7.7	7.6	7.9	7.9

obtains the best results. For Portuguese, BM13 and all versions of our system achieved similar informativeness scores, whereas our method generated significantly shorter compressions with an absolute decrease in the range 3.0–10.1 points. For the French corpus, it is complicated to define the best system because the second baseline, ILP:80% and ILP:∞ have similar informativeness scores for similar CRs. An inspection of the compressions generated by all systems highlighted that the low performance of our approach for the French dataset is partly related to the structure of negative sentences in French. In this language, these sentences must usually be composed of the tokens “ne” and “pas” to be correct, like in the following example: “La France n’a pas remporté le championnat du monde de volley-ball” (France did not win the world volleyball championship). In the studied dataset, the French corpus contains 27 negative source sentences divided into 13 clusters. Our approach often missed one of these tokens in its output compressions with the negative structure, which reduced the scores for informativeness and grammaticality. A post-processing of compressions could check if these two tokens are presented in the compression and correct this error.

Tables 7 and 8 show that the informativeness scores and keywords are related, i.e., the higher the number of keywords the higher the informativeness score. According to its type (with respect to the size and the amount of information), a cluster can have a different number of real keywords (more or less than 10 keywords). The number of keywords and informativeness scores are related, except for BM13 on the French corpus that used fewer keywords than our method and still generated more informative compressions.

The POS-LM post-processing does not improve significantly the compression quality of our method. This post-processing maintain or enhance grammaticality for all corpora, except for the ILP:∞+LM for Portuguese corpus, and informativeness for the Portuguese and the Spanish corpora. The biggest difference between these two versions of all methods is on the Spanish corpus (differences of 0.1 and 0.14 are observed for informativeness and grammaticality, respectively), for which the POS-LM version generated a longer version (CR is increased by 5.8 points), which justifies the improvement of informativeness.

Table 8. Manual evaluation of compression (ratings are expressed on a scale of 0 to 2). The best results are in bold (* and ** indicate significance at the 0.01 and the 0.001 level using ANOVA's test related to F10, respectively; † and †† indicate significance at the 0.01 and the 0.001 level using ANOVA's test related to BM13, respectively).

Metrics	Methods					
	F10	BM13	ILP:80%	ILP:80%+LM	ILP:∞	ILP:∞+LM
French						
Informativeness						
Score 0	20%	10%	14%	16%	14%	14%
Score 1	36%	31%	32%	35%	27%	34%
Score 2	44%	59%	54%	49%	59%	52%
Avg.	1.25 ± 0.8	$\mathbf{1.48 \pm 0.7}$	1.40 ± 0.7	1.33 ± 0.7	1.45 ± 0.7	1.39 ± 0.7
Grammaticality						
Score 0	6%	7%	12%	8%	10%	10%
Score 1	23%	29%	36%	29%	35%	36%
Score 2	71%	64%	52%	63%	55%	54%
Avg.	$\mathbf{1.65 \pm 0.6}$	1.56 ± 0.6	1.44 ± 0.7	1.55 ± 0.6	1.45 ± 0.7	1.44 ± 0.7
Portuguese						
Informativeness						
Score 0	9%	7%	8%	5%	7%	8%
Score 1	30%	16%	18%	22%	12%	13%
Score 2	61%	77%	74%	73%	81%	79%
Avg.	1.51 ± 0.7	1.70 ± 0.6	1.66 ± 0.6	1.68 ± 0.6	$\mathbf{1.74 \pm 0.6}$	1.71 ± 0.6
Grammaticality						
Score 0	9%	8%	6%	5%	4%	7%
Score 1	21%	18%	18%	21%	15%	17%
Score 2	70%	74%	76%	74%	81%	76%
Avg.	1.61 ± 0.6	1.66 ± 0.6	1.71 ± 0.6	1.69 ± 0.6	$\mathbf{1.76 \pm 0.5}$	1.68 ± 0.6
Spanish						
Informativeness						
Score 0	24%	26%	12%	11%	10%	10%
Score 1	49%	31%	39%	36%	39%	29%
Score 2	27%	43%	49%	53%	51%	61%
Avg.	1.02 ± 0.7	1.16 ± 0.8	1.36 ± 0.7 **	1.41 ± 0.7 **	1.40 ± 0.7 **	$\mathbf{1.50 \pm 0.7}$ **††
Grammaticality						
Score 0	11%	18%	12%	8%	10%	6%
Score 1	26%	33%	35%	36%	35%	29%
Score 2	63%	49%	53%	56%	55%	65%
Avg.	1.51 ± 0.7	1.30 ± 0.8	1.40 ± 0.7	1.48 ± 0.6	1.45 ± 0.7	$\mathbf{1.59 \pm 0.6}$ †

5.3 Applications

Most of previous MSC approaches have been applied on the Text Summarization problem and its variations. Among these works, several versions of our ILP method on different types of documents and in multiple languages have been successfully tested.

In the first application, the ILP approach was applied to the problem of microblog contextualization [17, 19]. Given a microblog about a festival, Linhares Pontes et al.'s [17, 19] system was able to generate a summary (maximum of 120 words) in four languages (English, French, Portuguese and Spanish) of Wikipedia's pages describing this microblog. In order to get more information about these festivals, they used Wikipedia to find information about these festivals and adapt the MSC method to extract relevant information related to the festival and generate a summary.

Linhares Pontes et al. [15] also investigated the generation of cross-lingual speech summaries of news documents. The goal was to analyze an audio file in French and generate a text summary in English. Contrary to the text document, the transcription of audio files must use Automatic Speech Recognition (ASR), which complicates and reduces the quality of the summary generation.

They adapted the MSC method to analyze sentences, both in their original and translated forms, and generate informative compressions in English using the relevance of French and English sentences. Their MSC method also analyzed 3 grams to add grammatically correct sequences of words into the compressions. This feature allowed their method to generate compressions with a good grammaticality, even when there are erroneous transcribed sentences.

Finally, Linhares Pontes et al. [18, 20, 21] also dealt with the issue of Cross-Language Text Summarization to generate English and French summaries from clusters of news documents in French, Portuguese and Spanish languages. Their MSC approach was applied on similar sentences among the documents to summarize. Despite the variety of these sentences (short, long, verbal and non-verbal sentences) and the introduction of errors by the used machine translation engine, experiments showed that the system usually generated correct compressions that are shorter and more informative than their source sentences.

6 Conclusion

Multi-Sentence Compression aims to generate a short informative text summary from several sentences with related and redundant information. Previous works built word graphs weighted by cohesion scores from the input sentences, then selected the best path to select words of the output sentence. We introduced in this study a model for MSC with two novel features. Firstly, we extended the work done by Boudin and Morin [5] that introduced keywords to post-process lists of N-best compressions. We proposed to represent keywords as labels directly on the vertices of word graphs to ensure the use of different keywords in the selected paths. Secondly, we devised an ILP modeling to take into account these new features with the cohesion scores, while selecting the best sentence. The compression ratio can be modulated with this modeling, by selecting for example a higher number of keywords for the sentences considered essential for a summary.

Our methodology was evaluated on three corpora built from Google news: a first one in French which had been built and used in [5], a second and a third one in Portuguese and in Spanish [22]. Automatic measures with the ROUGE package were supplemented with a manual evaluation carried out by human judges in terms of informativeness and grammaticality. We showed that keywords are important features to produce valuable compressed sentences. The paths selected with theses features generate results consistently improved in terms of informativeness while keeping up their grammaticality.

There are several potential avenues of work. We can use other kinds of language models based on Neural Networks [29] as an additional score to the optimization criterion to improve grammaticality. Another objective can be to manage polysemy through the use of the same label for the synonyms of each keyword inside the word graph. Finally, MSC can be jointly employed with the classical methods of Automatic Text Summarization by extraction in order to generate better summaries.

7 Appendix

Two examples in Spanish and Portuguese are provided in this section to illustrate the differences observed between the tested methods.

7.1 Spanish

The Spanish cluster (Table 9) is composed of 20 similar sentences. The vocabulary of this cluster is composed of 880 tokens and this cluster has a TTR of 33.3%. F10 generated the shortest compression; however, the sentence has missing information. The second baseline system and our method without post-processing generated incorrect compressions. Our method without post-processing generated a sentence with relevant keywords but it is not correct. The post-processing selected a more grammatical compression without reducing informativeness. The top 10 keywords selected by LDA were : *vuelo*, *cuba*, *fort*, *lauderdale*, *unidos*, *primer*, *jetblue*, *comercial*, *clara* and *florida*.

Table 9. Example in Spanish showing the first 3 sentences among 20 source sentences and 1 of 3 available references.

Source document	
El vuelo 387 de la aerolínea estadounidense JetBlue inauguró una nueva era en el transporte entre ambos países, al partir desde Fort Lauderdale (Florida, sureste) cerca de las 10:00 locales (14H00 GMT), y llegar a Santa Clara, 280 Km al este de La Habana, a las 10:57. (*Flight 387 of the US airline JetBlue inaugurated a new era in transport between the two countries, departing from Fort Lauderdale (Florida, southeast) at around 10:00 local time (14H00 GMT), and arriving in Santa Clara, 280 km east of Havana, at 10:57.*) Un avión de pasajeros de la línea aérea JetBlue despegó este miércoles a Cuba desde el aeropuerto Internacional de Fort Lauderdale en lo que viene a ser el primer vuelo regular entre Estados Unidos y la isla caribeña desde 1961, en un nuevo hito en la nueva fase de relaciones entre Washington y La Habana. (*A JetBlue airliner took off for Cuba on Wednesday from Fort Lauderdale International Airport, thus becoming the first regular flight between the United States and the Caribbean island since 1961, as a new milestone in the new phase of relations between Washington and Havana.*) La aerolínea JetBlue inaugurará los vuelos directos comerciales el 31 de agosto con un viaje entre Fort Lauderdale, Florida, hasta el aeropuerto de Santa Clara, a unos 270 kilómetros al este de La Habana, reportó la compañía estadunidense. (*JetBlue will inaugurate direct commercial flights on Aug. 31 with a trip from Fort Lauderdale, Florida, to Santa Clara airport, some 270 kilometers east of Havana, the U.S. company reported.*)	
Reference	
La aerolínea JetBlue Airways Corp inauguró el 31 de agosto los vuelos directos entre Estados Unidos y Cuba tras 50 años de suspensión . *(The airline JetBlue Airways Corp opened on August 31 direct flights between the United States and Cuba after 50 years of suspension .)*	
Compressions	
F10:	la aerolínea jetblue inauguró este miércoles a cuba el primer vuelo inaugural . *(the airline jetblue opened the inaugural first flight to cuba this wednesday .)*
BM13:	el aeropuerto de fort lauderdale , florida , sureste de estados unidos y cuba desde 1961 partió este miércoles el primer vuelo inaugural . *(the airport of fort lauderdale , florida , southeastern united states and cuba since 1961 departed this Wednesday on the inaugural first flight .)*
ILP:80%	el aeropuerto de fort lauderdale , florida , sureste de estados unidos y cuba desde 1961 partió este miércoles el primer vuelo inaugural . *(the airport of fort lauderdale , florida , southeastern united states and cuba since 1961 departed this Wednesday on the inaugural first flight .)*
ILP:80%+LM	la aerolínea jetblue inauguró este miércoles el primer vuelo desde fort lauderdale , florida , sureste de estados unidos a cuba desde 1961 . *(the airline jetblue opened Wednesday the first flight from fort lauderdale , florida , southeastern united states to cuba since 1961.)*
ILP:∞	el aeropuerto de fort lauderdale , florida , sureste de estados unidos y cuba desde 1961 partió este miércoles el primer vuelo inaugural . *(the airport of fort lauderdale , florida , southeastern united states and cuba since 1961 departed this Wednesday on the inaugural first flight .)*
ILP:∞+LM	la aerolínea jetblue inauguró este miércoles el primer vuelo desde fort lauderdale , florida , sureste de estados unidos a cuba desde 1961 . *(jetblue airlines inaugurated this wednesday the first flight from fort lauderdale, florida , southeastern united states to cuba since 1961 .)*

7.2 Portuguese

Table 10 displays a cluster composed of 11 Portuguese sentences with a TTR of 37% and a vocabulary of 351 tokens. In this case, F10 did not generate the shortest compression and has

incorrect information. The second baseline, which post-processes the outputs of the first one, was not able to correct the errors. Almost all versions of our method generated the shortest and the most informative compressions related to the text. Our method without post-processing generated the best compression. The post-processing selected a more grammatically correct sentence, while its information is incorrect. The top 10 keywords selected by LDA were : *tesla*, *solarcity*, *milhões*, *2,6*, *solar*, *empresa*, *carros*, *fabricante*, *dólares* and *motors*.

Acknowledgments

This work was partially financed by the European Project CHISTERA-AMIS ANR-15-CHR2-0001 and the European Union's Horizon 2020 research and innovation program under grants 770299 (NewsEye) and 825153 (EMBEDDIA).

References

1. **Banerjee, S., Mitra, P., & Sugiyama, K. (2015).** Multi-document abstractive summarization using ilp based multi-sentence compression. *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, AAAI Press, pp. 1208–1214.
2. **Barzilay, R. & Lapata, M. (2006).** Aggregation via set partitioning for natural language generation. *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL '06, Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 359–366.
3. **Barzilay, R. & McKeown, K. R. (2005).** Sentence fusion for multidocument news summarization. *Comput. Linguist.*, Vol. 31, No. 3, pp. 297–328.
4. **Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003).** Latent Dirichlet allocation. *J. Mach. Learn. Res.*, Vol. 3, pp. 993–1022.
5. **Boudin, F. & Morin, E. (2013).** Keyphrase extraction for n-best reranking in multi-sentence compression. *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, HLT-NAACL, The Association for Computational Linguistics, pp. 298–305.
6. **Bruckner, S., Hüffner, F., Komusiewicz, C., & Niedermeier, R. (2013).** *Evaluation of ILP-Based Approaches for Partitioning into Colorful Components*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 176–187.
7. **Clarke, J. & Lapata, M. (2007).** Modelling compression with discourse constraints. *Proceedings of the Conference on Empirical Methods in Natural Language Processing and on Computational Natural Language Learning (EMNLP-CoNLL-2007)*, Prague, Czech Republic, pp. 1–11.
8. **Clarke, J. & Lapata, M. (2008).** Global inference for sentence compression an integer linear programming approach. *J. Artif. Int. Res.*, Vol. 31, No. 1, pp. 399–429.
9. **Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990).** Indexing by latent semantic analysis. *Journal of the American society for Information Science*, Vol. 41, No. 6, pp. 391–407.
10. **Filippova, K. (2010).** Multi-sentence compression: Finding shortest paths in word graphs. *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 322–330.
11. **Filippova, K., Alfonseca, E., Colmenares, C. A., Kaiser, L., & Vinyals, O. (2015).** Sentence compression by deletion with lstms. **Màrquez, L., Callison-Burch, C., Su, J., Pighin, D., & Marton, Y.**, editors, *EMNLP*, The Association for Computational Linguistics, pp. 360–368.
12. **Filippova, K. & Strube, M. (2008).** Sentence fusion via dependency graph compression. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP, Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 177–185.
13. **Huet, S., Gravier, G., & Sébillot, P. (2010).** Morpho-syntactic post-processing of n-best lists for improved french automatic speech recognition. *Computer Speech and Language*, Vol. 24, No. 4, pp. 663–684.
14. **Lin, C.-Y. (2004).** Rouge: a package for automatic evaluation of summaries. *Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*.
15. **Linhares Pontes, E., González-Gallardo, C.-E., Torres-Moreno, J.-M., & Huet, S. (2019).** Cross-lingual speech-to-text summarization. **Choroś, K., Kopel, M., Kukla, E., & Siemiński, A.**, editors, *Mul-*

Table 10. Example in Portuguese showing the first 3 sentences among 11 source sentences and 1 of 2 available references.

Source document	
A Tesla fez uma oferta de compra à empresa de serviços de energia solar SolarCity por mais de 2300 milhões de dólares *(Tesla made an offer to purchase the SolarCity solar energy services company for over 2,300 million dollars.)*.	
A Tesla Motors, fabricante de carros elétricos, anunciou aquisição da SolarCity por US$ 2,6 bilhões *(Tesla Motors, a manufacturer of electric cars, announced the purchase of SolarCity for $2.6 billion.)*.	
A fabricante de carros elétricos e baterias Tesla Motors disse nesta segunda-feira (1) que chegou a um acordo com a SolarCity para comprar a instaladora de painéis solares por US$ 2,6 bilhões, em um grande passo do bilionário Elon Musk para oferecer aos consumidores um negócio totalmente especializado em energia limpa, informou a Reuters (*Electric car and battery manufacturer Tesla Motors said on Monday (1) that it reached an agreement with SolarCity to buy the solar panel installer for $2.6 billion, in a big step took by billionaire Elon Musk to offer consumers a fully specialized clean energy business, Reuters reported.*).	
Reference	
A Tesla Motors anunciou acordo para comprar a SolarCity por US$ 2,6 bilhões. (*Tesla Motors has announced an agreement to buy SolarCity for US$ 2.6 billion.*)	
Compressions	
F10	a solarcity para comprar a instaladora de painéis solares por us$ 2,6 bilhões *(solarcity to buy the solar panel installer for us$ 2.6 billions .)*
BM13	a solarcity para comprar a instaladora de painéis solares por us$ 2,6 mil milhões de dólares *(solarcity to buy the solar panel installer for us$ 2.6 billion dollars).*
ILP:80%	a tesla vai comprar a solar solarcity por 2,6 mil milhões de dólares *(tesla will buy the solar solarcity for 2.6 billion dollars.)*
ILP:80%+LM	a solarcity para comprar a instaladora de painéis solares por 2,6 mil milhões de dólares *(solarcity to buy the solar panel installer for 2.6 billion dollars.)*
ILP:∞	a tesla vai comprar a solar solarcity por 2,6 mil milhões de dólares *(tesla will buy the solar solarcity for 2.6 billion dollars.)*
ILP:∞+LM	a solarcity para comprar a instaladora de painéis solares por 2,6 mil milhões de dólares *(solarcity to buy the solar panel installer for 2.6 billion dollars.)*

timedia and Network Information Systems, Springer International Publishing, Cham, pp. 385–395.

16. **Linhares Pontes, E., Huet, S., Linhares, A. C., & Torres-Moreno, J.-M. (2018).** Multi-sentence compression with word vertex-labeled graphs and integer linear programming. *Proceedings of the 12th Workshop on Graph-Based Natural Language Processing (TextGraphs)*, Association for Computational Linguistics.

17. **Linhares Pontes, E., Huet, S., & Torres-Moreno, J.-M. (2018).** Microblog contextualization: Advantages and limitations of a multi-sentence compression approach. **Bellot, P., Trabelsi, C., Mothe, J., Murtagh, F., Nie, J. Y., Soulier, L., SanJuan, E., Cappellato, L., & Ferro, N.**, editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, Springer International Publishing, Cham, pp. 181–190.

18. **Linhares Pontes, E., Huet, S., & Torres-Moreno, J.-M. (2018).** A multilingual study of compressive cross-language text summarization. *Proceedings of the 17th Mexican International Conference on Artificial Intelligence (MICAI)*, Springer, Guadalajara, Mexico.

19. **Linhares Pontes, E., Huet, S., Torres-Moreno, J.-M., & Linhares, A. C. (2017).** Microblog contextualization using continuous space vectors: Multi-sentence compression of cultural documents. *Working Notes of the CLEF Lab on Microblog Cultural Contextualization*, volume 1866, CEUR-WS.org.

20. **Linhares Pontes, E., Huet, S., Torres-Moreno,**

J.-M., & Linhares, A. C. (2018). Cross-language text summarization using sentence and multi-sentence compression. **Silberztein, M., Atigui, F., Kornyshova, E., Métais, E., & Meziane, F.**, editors, *Natural Language Processing and Information Systems*, Springer International Publishing, Cham, pp. 467–479.

21. **Linhares Pontes, E., Huet, S., Torres-Moreno, J.-M., & Linhares, A. C. (2020).** Compressive approaches for cross-language multi-document summarization. *Data & Knowledge Engineering*, Vol. 125, 101763.

22. **Linhares Pontes, E., Torres-Moreno, J.-M., Huet, S., & Linhares, A. C. (2018).** A new annotated portuguese/spanish corpus for the multi-sentence compression task. *Proceedings of the 11th edition of the Language Resources and Evaluation Conference (LREC 2018)*.

23. **Luong, A., Tran, N., Ung, V., & Nghiem, M. (2015).** Word graph-based multi-sentence compression: Re-ranking candidates using frequent words. **Mérialdo, B., Nguyen, M. L., Le, D., Duong, D. A., & Tojo, S.**, editors, *2015 Seventh International Conference on Knowledge and Systems Engineering, KSE 2015, Ho Chi Minh City, Vietnam, October 8-10, 2015*, IEEE, pp. 55–60.

24. **McKeown, K., Rosenthal, S., Thadani, K., & Moore, C. (2010).** Time-efficient creation of an accurate sentence fusion corpus. *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 317–320.

25. **Miao, Y. & Blunsom, P. (2016).** Language as a latent variable: Discrete generative models for sentence compression. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, pp. 319–328.

26. **Mihalcea, R. & Tarau, P. (2004).** Textrank: Bringing order into texts. **Lin, D. & Wu, D.**, editors, *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP, Association for Computational Linguistics, Barcelona, Spain, pp. 404–411.

27. **Napoles, C., Van Durme, B., & Callison-Burch, C. (2011).** Evaluating sentence compression: Pitfalls and suggested remedies. *Proceedings of the Workshop on Monolingual Text-To-Text Generation*, MTTG '11, Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 91–97.

28. **Nayeem, M. T., Fuad, T. A., & Chali, Y. (2018).** Abstractive unsupervised multi-document summarization using paraphrastic sentence fusion. *Proceedings of the 27th International Conference on Computational Linguistics*, Association for Computational Linguistics, Santa Fe, New Mexico, USA, pp. 1191–1204.

29. **Niu, J., Chen, H., Zhao, Q., Su, L., & Atiquzzaman, M. (2017).** Multi-document abstractive summarization using chunk-graph and recurrent neural network. *Proceedings of the 2017 IEEE International Conference on Communications (ICC)*, pp. 1–6.

30. **Öncan, T., Altınel, İ. K., & Laporte, G. (2009).** A comparative analysis of several asymmetric traveling salesman problem formulations. *Computers & Operations Research*, Vol. 36, No. 3, pp. 637–654.

31. **Reape, M. & Mellish, C. (1999).** Just what is aggregation anyway? **Dizier, P. S.**, editor, *Proceedings of the 7th European Workshop on Natural Language Generation*, Toulouse, pp. 20–29.

32. **Rush, A. M., Chopra, S., & Weston, J. (2015).** A neural attention model for abstractive sentence summarization. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Lisbon, Portugal, pp. 379–389.

33. **Schmid, H. (1995).** Improvements in part-of-speech tagging with an application to German. *Proceedings of the ACL SIGDAT Workshop*, pp. 47–50.

34. **ShafieiBavani, E., Ebrahimi, M., Wong, R. K., & Chen, F. (2016).** An efficient approach for multi-sentence compression. **Durrant, R. J. & Kim, K.-E.**, editors, *Proceedings of The 8th Asian Conference on Machine Learning*, volume 63 of *Proceedings of Machine Learning Research*, PMLR, The University of Waikato, Hamilton, New Zealand, pp. 414–429.

35. **Shang, G., Ding, W., Zhang, Z., Tixier, A., Meladianos, P., Vazirgiannis, M., & Lorré, J.-P. (2018).** Unsupervised abstractive meeting summarization with multi-sentence compression and budgeted submodular maximization. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Melbourne, Australia, pp. 664–674.

36. **Thadani, K. & McKeown, K. (2013).** Supervised sentence fusion with single-stage inference. *Pro-*

ceedings of the Sixth International Joint Conference on Natural Language Processing, Asian Federation of Natural Language Processing, pp. 1410–1418.

37. **Torres-Moreno, J.-M. (2014).** *Automatic Text Summarization*. John Wiley & Sons.
38. **Tzouridis, E., Nasir, J. A., & Brefeld, U. (2014).** Learning to summarise related sentences. *Proceedings of the 25th International Conference on Computational Linguistics (COLING)*, ACL, pp. 1636–1647.
39. **Zhao, Y., Shen, X., Bi, W., & Aizawa, A. (2019).** Unsupervised rewriter for multi-sentence compression. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Florence, Italy, pp. 2235–2240.
40. **Zheng, C., Swenson, K., Lyons, E., & Sankoff, D. (2011).** OMG! orthologs in multiple genomes — competing graph-theoretical formulations. **Przytycka, T. M. & Sagot, M.-F.**, editors, *WABI*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 364–375.