Computing and Informatics, Vol. 32, 2013, 393–410

TEXT SEGMENTATION USING ROGET-BASED WEIGHTED LEXICAL CHAINS

Doina Tatar, Diana Inkpen, Gabriela Czibula

University "Babeş-Bolyai"
1 Kogalniceanu 400084
Cluj-Napoca, Romania
University of Ottawa
Ottawa, Canada
e-mail: dtatar@cs.ubbcluj.ro

Communicated by Jacek Kitowski

Abstract. In this article we present a new method for text segmentation. The method relies on the number of lexical chains (LCs) which end in a sentence, which begin in the following sentence and which traverse the two successive sentences. The lexical chains are based on Roget's thesaurus (the 1987 and the 1911 version). We evaluate the method on ten texts from the DUC 2002 conference and on twenty texts from the CAST project corpus, using a manual segmentation as gold standard.

Keywords: Lexical chains, text segmentation, topic boundaries, Roget's thesaurus, segmentation evaluation

1 INTRODUCTION

The purpose of linear text segmentation is to obtain groups of successive sentences which are linked to each other from a specified point of view. The segmentation is often a valuable stage in many natural language applications [23, 1]. For example, the identification of a passage in information retrieval is more useful than whole-document retrieval. A clear separation of the text into segments helps in summarization tasks, as argued in [3, 21].

Text segmentation methods rely on the cohesion of a text, regarded as "a device for sticking together different parts of the text" [2]. One way of identifying cohesion for automatic segmentation is by detecting the lexical chains in the text. Lexical

chains are sequences of words which are in a lexical cohesion relation with each other and tend to indicate portions of a text that form semantic units [20, 18, 24]; they could serve as a basis for segmentation and/or summarization [6]. Lexical chains can be constructed in a bottom-up manner [2, 4], by accumulated words that are related according to a thesaurus, dictionary, or other semantic relations. We chose an implementation of lexical chains based on Roget's thesaurus. We used the freely available 1911 version[1] and also the 1987 edition of Penguins Roget's Thesaurus of English Words and Phrases from Pearson Education. The latter has a bit wider coverage, since recent technical terms are missing from the 1911 version.

The main goal of this article is to present and evaluate our method for segmentation based on the Lexical Chains Distribution (LCD), when the lexical chains are high-quality ones based on Roget's thesaurus. The boundaries of the segments are calculated by LCD on the basis of the distribution of the start and end points of the lexical chains, and on the basis of the lexical chains that traverse a sentence. The method is applied to the automatically-obtained lexical chains and to the human-defined lexical chains, for ten documents from the DUC2002 competition and on twenty texts from the CAST project[2] corpus from the University of Woolverhampton [21]. The segmentations obtained by LCD are compared to a manual segmentation.

The article is structured as follows: Section 2 surveys previous work in linear text segmentation, with focus on the methods that use lexical chains. Section 3 presents the new LCD method of text segmentation. The experiments and the results are presented in Section 4. Comparison with related work is discussed in Section 5. We conclude and present directions for future work in Section 6.

2 RELATED WORK IN LINEAR SEGMENTATION

One of the first methods of linear text segmentation used reference chains [17]. Consider that all the chains of *antecedents – anaphors* of a text are $CHR_1, \ldots, CHR_n$. A chain CHR_i contains the occurrences of entities identified as antecedents for a given anaphor and also the occurrences of this anaphor. The principle for detecting a boundary between two segments is as follows: the most frequent pair of antecedent – anaphor (P) is changed at a boundary, and stays unchanged inside a segment. So, if the most frequent pair antecedent – anaphor for the sentences $S_1, \ldots S_i$, denoted by $P(S_1, \ldots S_i)$ is different from $P(S_1, \ldots, S_{i+1})$, then there is a boundary between the sentence S_i and the sentence S_{i+1}. Otherwise, S_i and S_{i+1} are in the same segment.

Another method for linear text segmentation is provided by the Centering Theory [7]. Let us consider that the *forward looking* centers (the syntactically-ranked entities introduced by a sentence) are calculated, and the most well-ranked center (the *preferred* center) is established for each sentence S_i. The principle for detecting

[1] http://rogets.site.uottawa.ca/

[2] http://www.clg.wlv.ac.uk/projects/CAST/corpus/listfiles.php

a boundary between two segments is as follows: the *preferred* center is changed at a boundary, and stays unchanged inside a segment. Therefore, if the *preferred* center of S_i denoted by $CP(S_i)$ is different from $CP(S_{i+1})$, then there is a boundary between the sentences S_i and S_{i+1}. Otherwise, S_i and S_{i+1} are in the same segment.

The most implemented and well-known method of topic segmentation is TextTiling [8, 23]. The article [8] describes a model of discourse structure based on the notion of topic shift, and an algorithm for subdividing expository texts into contiguous, non-overlapping subtopic segments (this is why the method is called TextTiling). As the author explains, "instead of undertaking the difficult task of attempting to define 'what a topic is', we should concentrate on describing what we recognize as topic shift". TextTiling assumes that a set of lexical items is in use during the course of a given subtopic, and when that subtopic changes, a significant proportion of the vocabulary changes as well [14]. The central idea of TextTiling consists in comparing adjacent blocks of text of fixed size. The more words the blocks have in common, the higher the lexical score at the gap (the boundary candidate) between them. If a low lexical score is preceded by and followed by a high lexical score, this is assumed to indicate a shift in the vocabulary that corresponds to a subtopic change (a boundary) [14]. The lexical score of a gap is calculated as the *cosine* of vectors associated to the adjacent block texts, where a vector contains the number of times each lexical item occurs in the corresponding text.

Another method, proposed in [30], is called logical segmentation, because the score of a sentence is the number of sentences of the text which are *entailed* by it. The scores form a structure that indicates how the most important sentences alternate with less important sentences. This structure organizes the text according to its logical content. Due to some similarities with TextTiling algorithm this method is called Logical TextTiling (LTT). Similarly to the topic segmentation [8], in logic segmentation the focus is on describing the shifting in information content in the discourse. Simply, a valley (a local minim) in the obtained graph (the logical structure) is a boundary between two segments. This is in accordance with the definition of a boundary as a perceptible discontinuity in the text structure [3], in this case a perceptible discontinuity in the logical flux of the sentences.

2.1 Linear Segmentation by Lexical Chains

Lexical chains (LCs) are sequences of words that are connected by semantic relations. The first work that used LCs (manually built) to indicate the structure of a text is [18], and it relies on the hierarchical structure of Roget's thesaurus to find semantic relations between words. In [29], a top-down method of linear text segmentation is proposed; it is based on the lexical cohesion of a text. Namely, first a single chain of disambiguated words in a text is established; then the rips of this chain are considered. These rips are boundaries of the segments in the cohesion structure of the text. Thus, a segment is considered as a piece of text where the disambiguation (the mining) of contained words is "chained". Due to some similarities with TextTiling algorithm, the method is called Cohesion TextTiling (CTT). The

chain of disambiguated words of a text is obtained by a Lesk-type algorithm [12] that uses definitions (glosses) from WordNet [13].

The work [29] compares segmentations by LTT and CTT, by comparing the summaries obtained applying the above strategies. The conclusion is that the quality of CTT summaries is higher than the quality of the LTT summaries from the point of view of informativeness [21].

3 TEXT SEGMENTATION USING LEXICAL CHAINS DISTRIBUTION

3.1 Extracting the Lexical Chains

Usually a lexical chain is obtained in a bottom-up fashion, by taking each candidate words from the text, and finding an appropriate relation offered by a thesaurus. Roget's thesaurus was used to build lexical chains in [18] and WordNet was used in [26]. If a semantic relation is found, the word is inserted with the appropriate sense in the current chain, and the senses of the other words in the chain are updated. If no chain is found, then a new chain is initiated.

We use Roget's thesaurus to detect relations between words. The thesaurus is structured into classes, sections, head groups, heads, parts of speech (if a word has more than one part of speech), paragraphs, and semi-colon groups. This organization starts from generic and loosely-related words down to very specific and highly similar (i.e., words in the same semi-colon groups are considered synonyms). An example of Roget entry is presented in Figure 1 for class VI, section III, head group 1, for the head **Love**. The parts of speech are noun and verb; there are several paragraphs for each; and finally the synonyms are in the semicolon groups.

We used the implementation of Jarmasz and Szpakowicz [9] for extracting lexical chains. In the first step, their method selects candidate words from the text, skipping stop-words and words that are not in the thesaurus. In the second step, the relations that are used to place words in the same chains are identified. The two main relations are repetition (repeated words, or different morphological variants of the same word), and inclusion in the same head in the thesaurus. Heads were chosen because they are neither too general, nor too specific. Words in the same head are about the same concept; for example: *bank* and *slope* in the same head, *Height* [9]. In the third step, the word is inserted into the chain. The next step merges some lexical chains and keeps only the stronger ones. The final lexical chains have associated weights that reflect the degree of similarity of the words from each chain, based on how far apart they are in the structure of the thesaurus and how many words are present in the chain. Therefore, we will use the extracted lexical chains, and also their weights.

Figure 2 shows an example of text from our dataset (the first text, AP880911), split into sentences. The lexical chains extracted from Roget 1911 are shown in Figure 3, with their associated weights.

VI. WORDS RELATING TO THE SENTIMENT AND MORAL POWERS
III. SYMPATHETIC AFFECTIONS
1. Social Affections
Love.
Nouns
love; fondness; liking; inclination (desire) [more]; regard, dilection, admiration, fancy.
affection, sympathy, fellow-feeling; tenderness; heart, brotherly love; benevolence [more]; attachment.
attractiveness; popularity; favorite [more].
...
Verbs
love, like, affect, fancy, care for, take an interest in, be partial to, sympathize with; affection; be in love with; have a love for, entertain a love for, harbor cherish a love for; regard, revere; take to, bear love to, be wedded to; set one's affections on; make much of, feast one's eyes on; hold dear, prize; hug, cling to, cherish, pet.
burn; adore, idolize, love to distraction, dote on, dote upon.
...

Fig. 1. Example of Roget entry for the head **Love** (shortened)

(S1) Hurricane Gilbert swept toward the Dominican Republic Sunday, and the Civil Defense alerted its heavily populated south coast to prepare for high winds, heavy rains and high seas. (S2) The storm was approaching from the southeast with sustained winds of 75 mph gusting to 92 mph. (S3) There is no need for alarm, Civil Defense Director Eugenio Cabral said in a television alert shortly before midnight Saturday. (S4) Cabral said residents of the province of Barahona should closely follow Gilbert's movement. (S5) An estimated 100,000 people live in the province, including 70,000 in the city of Barahona, about 125 miles west of Santo Domingo. (S6) Tropical Storm Gilbert formed in the eastern Caribbean and strengthened into a hurricane Saturday night. (S7) The National Hurricane Center in Miami reported its position at 2 am Sunday at latitude 16.1 north, longitude 67.5 west, about 140 miles south of Ponce, Puerto Rico, and 200 miles southeast of Santo Domingo. (S8) The National Weather Service in San Juan, Puerto Rico, said Gilbert was moving westward at 15 mph with a "broad area of cloudiness and heavy weather" rotating around the center of the storm. (S9) The weather service issued a flash flood watch for Puerto Rico and the Virgin Islands until at least 6 p.m. Sunday. (S10) Strong winds associated with the Gilbert brought coastal flooding, strong southeast winds and up to 12 feet to Puerto Rico's south coast. (S11) There were no reports of casualties. (S12) San Juan, on the north coast, had heavy rains and gusts Saturday, but they subsided during the night. (S13) On Saturday, Hurricane Florence was downgraded to a tropical storm and its remnants pushed inland from the U.S. Gulf Coast. (S14) Residents returned home, happy to find little damage from 80 mph winds and sheets of rain. (S15) Florence, the sixth named storm of the 1988 Atlantic storm season, was the second hurricane. (S16) The first, Debby, reached minimal hurricane strength briefly before hitting the Mexican coast last month.

Fig. 2. Example of text from the dataset, split into sentences

Lexical chains:
winds, storm, winds, storm, winds, winds, storm, storm, storm [weight: 9.25]
heavy, heavy, feet, feet, heavy [weight: 4.75]
south, north, south, south, north [weight: 4.25]
coast, coast, coast, coast [weight: 4.0]
rains, watch, rains, rain [weight: 3.25]
Sunday, Sunday, Sunday [weight: 3.0]
province, province, city [weight: 3.0]
miles, miles, miles [weight: 3.0]
hurricane, hurricane, hurricane [weight: 3.0]
night, strong, night, strength [weight: 3.0]
position, latitude, longitude [weight: 3.0]
alerted, alert, live [weight: 2.5]
center, issued, home [weight: 2.5]
seas, west, west [weight: 2.25]
estimated, reported, reports [weight: 2.25]
prepare, formed, brought [weight: 2.0]
National, National [weight: 2.0]
Service, service [weight: 2.0]
Juan, Juan [weight: 2.0]
flood, flooding [weight: 2.0]
approaching, moving [weight: 1.5]
gusting, gusts [weight: 1.5]
shortly, briefly [weight: 1.5]
flash, happy [weight: 1.5]
casualties, hitting [weight: 1.5]

Fig. 3. The lexical chains extracted from the text from Figure 2, using Roget 1911, with associated weights

3.2 Text Segmentation Using the Distribution of Lexical Chains

Let us consider the set of lexical chains described as:

$$LC_1 : [S_{i_1}, S_{j_1}], LC_2 : [S_{i_2}, S_{j_2}], \ldots$$

where S_{i_k} represents the first sentence containing a word of the lexical chain LC_k and S_{j_k} represents the last sentence containing a word of the lexical chain LC_k. Thus, for example LC_1 is represented as $[S_{i_1}, S_{j_1}]$ regardless the number of other sentences which contain words from LC_1 and are situated in the interior of this interval. Using the information about the first and the last sentence of each lexical chain, a score for each sentence can be calculated. Namely, let us denote by $input(S_i)$ the number of lexical chains which end in a sentence S_i, by $output(S_i)$ the number of lexical chains which begin in S_i, and by $during(S_i, S_{i+1})$ the number of lexical chains which traverse S_i (which is not a beginning of the lexical chain) and S_{i+1} (which is not an end of the lexical chain). We call this case a "strict traversal". The score of

Manual Lexical chains:

LC1:	Hurricane Gilbert (S1), Gilbert (S4), Gilbert (S8), Gilbert (S10)
LC2:	coast (S1), coast (S10), coast (S12), coast (S16)
LC3:	Civil Defense (S1), Civil Defense (S7)
LC4:	south (S1), southeast (S2) (part/whole), north (S7) (opposition), west (S7) (opposition), southeast (S10) (part/whole)
LC5:	winds (S1) (part/whole in relation to storm), rains (S1) (part/whole), seas (S1) (part/whole), storm (S2), winds (S2) , storm (S6) , hurricane (S6) (synonym), cloudiness (S8) (part/whole), weather (S8) (generalization), storm (S8), weather (S9), flood (S9) (part/whole), winds (S10), flooding (S10), winds (S10), rains (S12), gusts (S12) (specification), storm (S13), winds (S14), rains (S14), storm (S15), hurricane (S15) hurricane (S16)
LC6:	mph (S2); mph (S2); mph (S8); mph (S14)
LC7:	Eugenio Cabral (S3), Cabral (S4)
LC8:	alarm (S3) (generalization), alert (S3), watch (S9) (specification)
LC9:	midnight (S3) (specification), night (S6); night (S12)
LC10:	Sunday (S1) (part/whole), Saturday (S6) (part/whole), Sunday (S7), Sunday (S9), Saturday (S12), Saturday (S13)
LC11:	province (S4), province (S5) , city (S5) (part/whole)
LC12:	residents (S4), people (S5) (generalization), residents (S14)
LC13:	Barahona (S4), Barahona (S5)
LC14:	movement (S4), position (S7) (part/whole), area (S8) (part/whole), center (S8) (specification)
LC15:	Santo Domingo (S5), Santo Domingo (S7)

Fig. 4. Manual lexical chains extracted from the text from Figure 2

a sentence S_i, denoted by $score(S_i)$ is:

$$score(S_i) = \frac{input(S_i) + output(S_{i+1})}{during(S_i, S_{i+1})}$$

The justification of this formula is: if $input(S_i)$ and/or $output(S_{i+1})$ are big, then there is a good chance for S_i to be the final sentence of a segment, and for S_{i+1} to be the first sentence of the next segment. Also, if $during(S_i, S_{i+1})$ is small, this chance is increased, because a low number of lexical chains which "strictly traverse" two sentences indicates a weak link between them. So, the higher the $score(S_i)$, the higher the chance to have a boundary between S_i and S_{i+1}. In this way, the points of local maxima in the graph for all the numbers $score(S_i)$ indicate the boundaries of the segments of the text $S_1, \ldots, S_n$. Oppositely, the points of local minima in the graph of $score(S_i)$ display the sentences which are *most interior* (with a high number of lexical chains that "strictly traverse it") in a segment. The first-mentioned points are used in segmentation and both of them could be used in summarization [28].

The above formula improves a scoring method of [20, 26, 15] by introducing the term $during(S_i, S_{i+1})$ and by splitting the functions of a sentence to be a final

sentence or a first sentence in a segment. Let us mention that the improvement in segmentation performance by introducing the term $during(S_i, S_{i+1})$ was argued by us in some preliminary experiments.

The formula permits also the use of a number of h different weights for lexical chains. Let us remember that a lexical chain LC_k is represented as $[S_{i_k}, S_{j_k}]$ regardless the number of other sentences which contain words from LC_k and are situated in the interior of this interval. To express this information about a lexical chain the weight could be a solution. Weights for lexical chains could be their length, their density (the ratio between the number of total terms and the number of distinct terms in a chain), etc. Using different schemes for the weights could help in the study of the influence of different features: it is known that the question of preferring long or short chains for the study of the cohesion is not yet solved.

The set of lexical chains is described now as:

$$LC_1 : [S_{i_1}, S_{j_1}]; w_p^1, LC_2 : [S_{i_2}, S_{j_2}]; w_p^2, \ldots$$

where $p = 1, \ldots h$, and h is the number of weights for a lexical chain.

The functions:

$$input(S_i), output(S_i) \text{ and } during(S_i, S_{i+1})$$

are modified into

$$input(S_i)^p \text{ and } output(S_i)^p \text{ and } during(S_i, S_{i+1})^p, \quad p = 1, \ldots, h,$$

as follows:

$$\begin{aligned} input(S_i)^p &= \sum_{LC_j \text{ ends in } S_i} w_p^j \\ output(S_i)^p &= \sum_{LC_j \text{ begins in } S_i} w_p^j \\ during(S_i, S_{i+1})^p &= \sum_{LC_j \text{ strict traverses } S_i S_{i+1}} w_p^j. \end{aligned}$$

The above formula for scoring a sentence is slightly modified by adding the weight p:

$$score(S_i)^p = \frac{input(S_i)^p + output(S_{i+1})^p}{during(S_i, S_{i+1})^p}.$$

The points of local maxima in the graph of $score(S_i)^p$ indicate the boundaries of the text $S_1, \ldots, S_n$ when the weight p is considered.

Let us mention that in our experiments we used a single weight ($h = 1$) for each lexical chain. The weight assigned to a chain reflects the importance and the strength of the chain. It is obtained from Roget's thesaurus based on the degree of similarity between the words in the chain, based on how far apart they are in the structure of the thesaurus and how many words are present in the chain.

4 EXPERIMENTS AND EVALUATION

The validity of our LCD text segmentation method is proved in an experiment on ten texts from DUC2002 and on twenty texts from CAST project corpus, for five types of lexical chains. For each text, the sets of lexical chains are:

- lexical chains manually obtained (as gold standard), using all types of lexical relations;
- lexical chains obtained using Roget 1987;
- lexical chains obtained using Roget 1987, where the weights are also considered;
- lexical chains obtained using Roget 1911;
- lexical chains obtained using Roget 1911, where the weights are also considered.

The segmentations obtained from these sets of lexical chains by LCD method are denoted in short as Man, Roget 1911, Roget 1911w, Roget 1987 and Roget 1987w.

As evaluation measure we used the relative correctness of the segmentation method, calculated using the *WindowDiff* measure introduced in [22]. This measure is more appropriate than a "classical" measure of correctness such as precision or recall, that checks if the boundaries were determined with exact positions, because the latter measures would penalize "far-misses" and "near-misses" boundaries in the same way [22]. *WindowDiff* is an error measure which counts how many discrepancies occur between the reference and the system results:

$$WindowDiff(Hyp, Ref) = \frac{\sum_{i=1}^{N-k} \mid r(i,k) - h(i,k) \mid}{N-k}.$$

Here $r(i,k)$ represents the number of boundaries of the reference segmentation Ref contained between sentences i and $i+k$ and $h(i,k)$ represents the number of boundaries of the hypothesis segmentation Hyp contained between the sentences i and $i+k$. N is the total number of sentences. The selected value for k is 0. The correctness is calculated as:

$$Correctness(Hyp, Ref) = (1 - WindowDiff(Hyp, Ref)) \times 100\,\%. \quad (1)$$

We needed manually-built reference segmentations for the 30 texts, in order to be able to evaluate our LCD segmentation method. The manual segmentation was built by a professional linguist. We call this completely manual segmentation ManSeg. It considered only the texts, without looking at any lexical chains.

In addition, we manually determined the lexical chains. These chains were determined by a team of students under the supervision of the linguist, and the cases of disagreement were resolved.

Table 1 displays the correctness of segmentations, compared pair to pair for each text and averaged over the 10 first texts. LCD segmentations based on Roget

lexical chains and the LCD segmentation based on manual lexical chains (Man) are considered here, using the following denotations: the correctness, C, of the LCD segmentation based on Roget 1911, C(Roget 1911, Man); the same when the weights of the lexical chains are used in the calculations, C(Roget 1911w, Man); the correctness of the LCD segmentation based on Roget 1987 chains, C(Roget 1987, Man); and finally, the previous one when taking into account the weights of the lexical chains, C(Roget 1987w, Man).

From this table, we conclude that using the weights in the calculation improves the correctness of the segmentation. Using different versions of the thesaurus leads to different results, but our conclusion is that the correctness score is not higher when using Roget 1987 than when using Roget 1911.

Then we compared the manually-obtained segmentation ManSeg with the segmentation obtained with the LCD method applied to Roget 1987 lexical chains, to Roget 1987 lexical chains with weights, to Roget 1911 lexical chains, and to Roget 1987 lexical chains with weights. The results are presented in Table 2. From this table, again, we conclude that the weights improve the correctness.

For the example presented in Figure 2, the LCD segmentation obtained with the Roget 1911 lexical chains with weights (Figure 3) contains the following segments, where each segment is indicated by its first and last sentence: Segment 1: [1, 5]; Segment 2: [6, 7]; Segment 3: [8, 10]; Segment 4: [11, 12]; and Segment 5: [13, 16]. For the same text, the manual segmentation contains the following segments: Segment 1: [1, 2]; Segment 2: [3, 5]; Segment 3: [6, 12]; and Segment 4: [13, 16]. In this case the algorithm obtained a larger number of segments compared to the manual segmentation (5 instead of 4). The first two segments from the manual segmentation were put together into one segment, the second segment was split into three segments, and the last segment was correctly detected.

In Figure 5, we show the LCD segmentations that use different lexical chains, the gold manual segmentation and the TextTiling segmentation [8], all these for a single text.

Roget 1911 lexical chains: $[1,3][4,5][6,7][8,10][11,12][13,14][15,16]$
Roget1911 lexical chains with weights: $[1,5][6,7][8,10][11,12][13,16]$
Roget 1987 lexical chains: $[1,4][5,8][9,14][15,16]$
Roget 1987 lexical chains with weights: $[1,4][5,8][9,16]$
Manual lexical chains: $[1,3][4,7][8,10][11,12][13,14][15,16]$
Manual segmentation: $[1,2][3,5][6,12][13,16]$
TextTiling segmentation: $[1,3][4,9][10,16]$

Fig. 5. Different LCD segmentations obtained with the Roget 1911 and Roget 1987 lexical chains (with and without weights) for the text AP880911

To look at one example of measuring the LCD segmentation performance, the value 63 % of the *Correctness*(*Roget1911w*, *Man*) in Table 1 for the text AP880911 is calculated considering the vector 1000111101111001, corresponding to the LCD

segmentation using Roget 1911 based lexical chains with weights (see the second row of Figure 5), and the vector 1110110000011001 corresponding to the LCD segmentation using manual lexical chains (see the fifth row of Figure 5). Applying the formula for $WindowDiff$ with $k = 0$ for these two segmentations result in counting the different values of the two vectors and dividing by the number of sentences: in this case $6/16 = 0.37$. The correctness calculated with Equation (1) provides the value of 63 %.

A conclusion from Figure 5 is that the weighted lexical chains provide fewer segments than the cases without weights. Compare 7 segments obtained with the Roget 1911 lexical chains with 5 segments obtained with the Roget 1911 weighted lexical chains, or 4 and 3 segments for the Roget 1987 corresponding cases. Also, all the boundaries presented in a weighted case are contained in the set of boundaries for the un-weighted case: for example, in the Roget 1911 case, the set of last sentences $\{5, 7, 10, 12, 16\}$ is contained in the set $\{3, 5, 7, 10, 12, 14, 16\}$. The conclusion is that the weighted case provides a more condensed view of the segmentation.

From both Table 1 and Table 2, we observe that the segmentation that uses the Roget 1911 lexical chains is better than the segmentation that uses the 1987 version. This is contrary to the expectations anticipating that the older version of Roget obtains a bit lower performance because it might miss some recent terms that could occur in our test data. We analyzed some of the segmentations and lexical chains, in order to understand the differences. We noticed that the lexical chains built from Roget 1911 are more cohesive and this seems to help the segmentation. For the LCD segmentation method only the beginnings and the ends of the lexical chains matter. For example, in seventh text, FT9235589, the last chain from Roget 1911 is: "*pressures, pressure*" (from Sentence 12 to Sentence 20), while from Roget 1987, the words *pressures*, and *pressure* appear in the chain "*scale, convince, compelling, prompts, pressures, convincing, weight, pressure, people*" (Sentence 1 to Sentence 25). The first chain is more cohesive than the second one. In another example, from the same text, a chain obtained from Roget 1911 is: "*building, cut, cuts, construction, construction*" (Sentence 5 to Sentence 26), while from Roget 1987, the word *construction* appears in the chain "*industrial, companies, construction ...*" (Sentence 2 to Sentence 26) and the word *building* in the chain "*yields, erected, ...*" (Sentence 2 to Sentence 24). So, two words that should be in the same chain (*building* and *construction*) ended up in two different chains.

We also noticed that sometimes a word can end up in two chains, possibly because it has more than one sense used in the same document. This might confuse the segmentation. The method that uses the weights of the lexical chains performs better in such cases, because one of the chains (for the weaker sense) has lower weight.

The fact that Roget 1911 is very good for the task is an advantage, because this version of Roget's thesaurus is freely available to researchers.

Doc	C(R-87, Man)	C(R-87w, Man)	C(R-11, Man)	C(R-11w, Man)
*AP*880911	50 %	50 %	75 %	63 %
*AP*891018	78 %	71 %	78 %	63 %
*AP*890922	53 %	62 %	53 %	72 %
*AP*880314	56 %	56 %	78 %	56 %
*AP*880817	43 %	50 %	65 %	72 %
*AP*890323	66 %	66 %	59 %	66 %
*FT*9235589	54 %	47 %	62 %	62 %
*AP*900621	64 %	64 %	46 %	46 %
*AP*890925	50 %	60 %	55 %	60 %
*AP*900103	70 %	70 %	54 %	70 %
Average	58.4 %	59.6 %	62.5 %	63.1 %

Table 1. The segmentation correctness relative to the segmentation based on manual lexical chains (Man) for different Roget-based lexical chains

Doc	C(R-87, ManSeg)	C(R-87w, ManSeg)	C(R-11, ManSeg)	C(R-11w, ManSeg)
*AP*880911	38 %	50 %	63 %	63 %
*AP*891018	41 %	41 %	49 %	54 %
*AP*890922	43 %	53 %	53 %	53 %
*AP*880314	100 %	100 %	56 %	56 %
*AP*880817	50 %	50 %	72 %	72 %
*AP*890323	43 %	49 %	55 %	55 %
*FT*9235589	47 %	54 %	77 %	70 %
*AP*900621	63 %	46 %	64 %	64 %
*AP*890925	43 %	37 %	41 %	46 %
*AP*900103	54 %	54 %	39 %	39 %
Average	52.5 %	53.4 %	56.9 %	57.2 %

Table 2. The LCD segmentation correctness relative to the manual segmentation (ManSeg) for different Roget-based lexical chains

5 COMPARISON TO RELATED WORK

It is interesting to compare, on the one hand, our LCD method with the "classical" and intensely performed TextTiling method when a manual segmentation ManSeg of the corpus of 30 texts is considered as gold standard. On the other hand, we applied the LCD method to the lexical chains obtained using WordNet and compared the resulting segmentation with the manual segmentation ManSeg. The results are given in Tables 3 and 4.

The first column in both tables contains the results of our LCD segmentation method based on lexical chains computed from WordNet, using a Word Sense Disambiguation algorithm to choose the appropriate WordNet senses [13]. The se-

cond column contains the results of our re-implementation of the TextTiling algorithm [8].

Doc	C(WN Chains, ManSeg)	C(TextTiling, ManSeg)	C(R-11w, ManSeg)
*AP*880911	29 %	50 %	63 %
*AP*891018	56 %	75 %	54 %
*AP*890922	27 %	53 %	53 %
*AP*880314	72 %	56 %	56 %
*AP*880817	54 %	65 %	72 %
*AP*890323	52 %	60 %	55 %
*FT*9235589	63 %	47 %	70 %
*AP*900621	78 %	46 %	64 %
*AP*890925	53 %	46 %	46 %
*AP*900103	46 %	39 %	39 %
Average	53.0 %	53.7 %	57.2 %

Table 3. Comparison between our best method and related methods for DUC2002 documents (segmentation correctness relative to the manual segmentation ManSeg)

We note that our method based on Roget 1911 with weighted lexical chains achieves the best results both for the first 10 texts and for the last 20 texts. This fact is in concordance with the reported difficulties in word sense disambiguation using WordNet [31]. Our better result for Roget could be explained by the underchaining phenomenon of LCs building using WordNet signaled in [27, 25, 5]: the lack of connections and the lack of consistency in the semantic proximity. An example in [5] shows that the words "*blind*" and "*rainbow*" have an intuitive association concerned with "*sight*" and "*visual phenomena*", that is reflected in their membership in the same group of Roget categories, but is not possible with WordNet. The work [16] experimentally proves that the traditional edge counting approach of semantic similarity works better in Roget than in WordNet, the end result being the ability to compare the relative usefulness of Roget and WordNet for different types of tasks.

We are aware that a corpus of 30 texts is too small to draw statistically valid conclusions. However, if LCD seems to be better in Tables 3 and 4, its strength could be emphasized by using different schemes for weights in the study of different features of LCs. On the other hand, we know that simple comparison of methods' mean accuracies is insufficient. In this respect we realized an analysis of variance (Anova) by calculating the ratio F for pairs of methods:

1. LCD with Roget and LCD with WordNet,
2. LCD with Roget and TextTiling.

It is known that if F is close to 1, then the between groups variance is similar to the within groups variance and the null hypothesis (that the groups do not differ

Doc	C(WN Chains, ManSeg)	C(TextTiling, ManSeg)	C(R-11w, ManSeg)
$T1.472239$	0.74 %	0.57 %	0.52 %
$T2.472294$	0.52 %	0.60 %	0.52 %
$T3.472295$	0.27 %	0.70 %	0.70 %
$T4.472296$	0.81 %	0.62 %	0.81 %
$T5.472297$	0.67 %	0.47 %	0.60 %
$T6.472303$	0.75 %	0.77 %	0.56 %
$T7.472339$	0.66 %	0.65 %	0.57 %
$T8.472355$	0.48 %	0.83 %	0.57 %
$T9.472364$	0.5 %	0.52 %	0.62 %
$T10.472415$	0.75 %	0.58 %	0.66 %
$T11.472439$	0.73 %	0.37 %	0.64 %
$T12.476501$	0.49 %	0.41 %	0.56 %
$T13.472451$	0.59 %	0.42 %	0.59 %
$T14.472455$	0.58 %	0.57 %	0.69 %
$T15.472462$	0.54 %	0.61 %	0.62 %
$T16.472474$	0.5 %	0.60 %	0.70 %
$T17.472617$	0.48 %	0.68 %	0.49 %
$T18.472475$	0.38 %	0.75 %	0.68 %
$T19.472559$	0.40 %	0.56 %	0.59 %
$T20.472567$	0.40 %	0.73 %	0.57 %
Average	56.60 %	60.05 %	61.30 %

Table 4. Comparison between our best method and related methods for the CAST documents (segmentation correctness relative to the manual segmentation ManSeg)

significantly from each other) holds [19]. For the first pair of samples (LCD with Roget and LCD with WordNet) the F ratio is 1.0963 and for the second pair (LCD with Roget and LCD with TextTiling) the F ratio is 1.0833.

Even though the Anova test does not strongly show that our method that uses the Roget lexical chains is significantly better than the segmentation results when using TextTiling or when using our method with WordNet-based chains, we observe that our results are consistently better in average on both the dataset of 10 documents from DUC 2010 and on the dataset of 20 CAST documents.

6 CONCLUSION AND FURTHER WORK

The contributions of this article consist in a novel method for text segmentation that used the distribution of the lexical chains, using Roget-based lexical chains.

We argued that the use of lexical chains distribution could be a powerful tool for text segmentation. The evaluation was realized from three different points of view: firstly, comparing LCD with TextTiling, the result is better in the first case. Secondly, comparing LCD which uses LCs obtained from Roget with the LCD which

uses LCs obtained from WordNet, the results are better in the first case, too. Finally, comparing LCD which uses weighted LCs and un-weighted LCs, the results are also better in the first case. All comparisons are made relative to manual segmentations of the texts.

In future work, we intend to study the influence of semantic relations used in the lexical chains, and using different weights for various types of relations. We also intend to study a clustering approach of topical linear and hierarchical segmentation using the methods presented in [10] and [11]. The similarity between two clusters could be simply the number of lexical chains with the origin in the first cluster and with the end in the second one.

Another direction of future work is to apply our LCD formula for scoring the sentences using reference chains instead of LCs.

Acknowledgements

We thank Mario Jarmasz and Stan Szpakowicz for making available their software for building lexical chains, and Alistair Kennedy for running this software on our data. We thank Martin Scaiano for re-implementing the TextTiling algorithm. We thank Emma Tamaianu-Morita for the manual segmentation and for supervising the creation of the manual lexical chains.

REFERENCES

[1] Allen, J.: Natural Language Understanding. 2nd ed., Benjamin/Cummings Publ. 1995.

[2] Barzilay, R.—Elhadad, M.: Using Lexical Chains for Text Summarization. In: Mani, J. and Maybury, M. (Eds.): Advances in Automated Text Summarization, MIT Press 1999.

[3] Boguraev, B.—Neff, B.: Lexical Cohesion, Discourse Segmentation and Document Summarization. Proceedings of the 33rd Hawaii International Conference on System Sciences 2000.

[4] Doran, W.—Stokes, N.—Carthy, J.—Dunnion, J.: Assesing the Impact of Lexical Chain Scoring Methods and Sentence Extraction Schemes on Summarization. Proceedings of the 5th International Conference on Intelligent Text Processing and Computational Linguistics 2004.

[5] Ellman, J.: Using Roget's Thesaurus to Determine the Similarity of Texts. Ph. D. thesis, University of Sunderland, June 2000.

[6] Ercan, G.—Cicekli, I.: Lexical Cohesion-Based Topic Modeling for Summarization. Proceedings of CICling 2008, pp. 582–592.

[7] Grosz, B.—Joshi, B.—Weinstein, S.: Centering: A Framework for Modelling the Local Coherence of Discourse. Computational Linguistics, Vol. 21, 1995, No. 2, pp. 203–225.

[8] Hearst, M.: TextTiling: Segmenting Text into Multi-Paragraph Subtopic Passages. Computational Linguistics 1997, pp. 33–64.

[9] Jarmasz, M.— Szpakowicz, S.: Roget's Thesaurus and Semantic Similarity. In Proceedings of Recent Advances in Natural Language Processing (RANLP) 2003, Bulgaria 2003, pp. 111–120.

[10] Kuta, M.—Wojcik, W.—Wrzeszcz, M.—Kitowski, J.: Application of Weighted Voting Taggers to Language Described with Large Tagsets. Computing and Informatics, Vol. 29, 2010, No. 2, pp. 203–225.

[11] Kuta, M.—Kitowski, J.: Benchmarking High Performance Architectures with Natural Language Processing Algorithms. Computer Science, Vol. 12, 2011, pp. 20–31.

[12] Lesk, M.: Automatic Sense Disambiguation using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. In Proceedings of the SIGDOC Conference 1986.

[13] Miller, G.: WordNet: A Lexical Database. Communications of the ACM, Vol. 38, 1997, pp. 39–41.

[14] Manning, C.—Schutze, H.: Foundation of Statistical Natural Language Processing. MIT 1999.

[15] Marathe, M.—Hirst, G.: Lexical Chains Using Distributional Measures of Concept. CICLing 2010, LNCS 6008, pp. 291–302.

[16] McHale, M.: A Comparison of WordNet and Roget's Taxonomy for Measuring Semantic Similarity. `http://xxx.lanl.gov/cmp-lg/9809003`, 14 Sept. 1998.

[17] Mitkov, R.: Anaphora Resolution. Pearson Education, Longman 2002.

[18] Morris, J.—Hirst, G.: Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text. Computational Linguistics, Vol. 17, 1991, No. 1, pp. 21–48.

[19] Oakes, M.: Statistics for Corpus Linguistics. Edinburgh University Press 1998.

[20] Okumura, M.—Honda, T.: WSD and Text Segmentation Based on Lexical Cohesion. Proceedings of COLING 1994, pp. 755–761.

[21] Orasan, C.: Comparative Evaluation of Modular Automatic Summarization Systems using CAST. Ph. D. Thesis, University of Wolverhampton, UK 2006.

[22] Pevzner,L.—Hearst, M.: A Critique and Improvement of an Evaluation Metric for Text Segmentation. Computational Linguistics, Vol. 28, 2002, No. 1, pp. 19–36.

[23] Reynar, J.: Topic Segmentation: Algorithms and Applications. Ph. D. Thesis, Univ. of Pennsylvania 1998.

[24] Silber, H.—McCoy, K.: Efficiently-Computed Lexical Chains, as an Intermediate Representation for Automatic Text Summarization. Computational Linguistics, Vol. 28, 2002, No. 4, pp. 487–496.

[25] Stairmand, M.—Black, W. J.: Conceptual and Contextual Indexing Using WordNet-Derived Lexical Chains. In Proceedings of BCS IRSG 1996.

[26] Stokes, N.: Spoken and Written News Story Segmentation using Lexical Chains. Proceedings of HLT-NAACL 2003, pp. 49–54.

[27] StOnge, D.: Detecting and Correcting Malapropisms with Lexical Chains. Technical Report CSRI-319, University of Toronto, Department of Computer Science, March 1995.

[28] Tatar, D.—Tamaianu-Morita, E.—Czibula, G.: Segmenting Text by Lexical Chains Distribution. Proceedings of the International Conference on Knowledge Engineering, Principles and Techniques (KEPT) 2009, Cluj-Napoca, Romania 2009, pp. 33–37.

[29] Tatar, D.—Mihis, A.—Serban, G.: Lexical Chains Cohesion Segmentation and Summarization. SYNASC 2008, Timisoara, IeAT Technical Report 2008, pp. 99–106.

[30] Tatar, D.—Mihis, A.—Lupsa, D.: Text Entailment for Logical Segmentation and Summarization. 13th International Conference on Applications of Natural Language to Information Systems, London, UK, LNCS 5039, 2008, pp. 233–244.

[31] Voorhees, E.: Query Expansion Using Lexical-Semantic Relations. Proceedings of SIGIR '94, pp. 61-69.

Doina Tatar is a Professor at the Faculty of Mathematics and Computer Science, Babes-Bolyai University, Cluj-Napoca, Romania. Her teaching and research interests concentrated on natural language processing, namely semantic and pragmatic aspects; she is also interested in automated theorem proving. During her long career she elaborated over a hundred of papers and books in the above research fields. She also serves as a Program Committee member of the international conferences SYNASC (Symposium on Symbolic and Numeric Algorithms for Scientific Computing), and KEPT (Knowledge Engineering: Principles and Techniques), and as an Editorial Board member of International Journal of Organizational and Collective Intelligence and International Journal on Advances in Information Sciences and Service Sciences.

Diana Inkpen obtained her Ph. D. in 2003 from the University of Toronto, Department of Computer Science. She obtained her B. Eng. from the Department of Computer Science, Technical University of Cluj-Napoca, Romania, in 1994, and a M. Sc. from the same university, in 1995. In 2003, after finishing her Ph. D., she joined the School of Electrical Engineering and Computer Science at the University of Ottawa, as an Assistant Professor; then she was Associate Professor till April 2012, and she is now a Professor. She was an Erasmus Mundus Visiting Scholar at the University of Wolverhampton, UK in 2009, and was named a Visiting Professor from September 2010. She published 7 book chapters, 21 journal papers and 78 referred conference and workshop papers. She organized 5 international workshops; she is a program co-chair for AI 2012. She is an associate editor of the Computational Intelligence journal. She had many research grants from NSERC, SSHRC, and OCE, including industrial collaborations. Her research work is in the area of compu-

tational linguistics, more specifically: analysis and generation of emotion in texts, lexical semantics, automatic text classification, and information retrieval.

Gabriela Czibula is Professor at the Department of Computer Science, Faculty of Mathematics and Computer Science, Babeş-Bolyai University of Cluj-Napoca, Romania. She has received her Ph. D. degree in computer science in 2003, with the "cum laude" distinction. She published more than 140 papers in prestigious journals and conference proceedings. Her research interests include artificial intelligence, machine learning, multiagent systems, programming paradigms.