Igor A. Bolshakov and A. Gelbukh. *Text segmentation into paragraphs based on local text cohesion.* In: Text, Speech and Dialogue (TSD-2001). Lecture Notes in Artificial Intelligence, N 2166, Springer-Verlag, 2001, pp. 158–166.

Text Segmentation into Paragraphs Based on Local Text Cohesion

Igor A. Bolshakov and Alexander Gelbukh

Center for Computing Research (CIC)
National Polytechnic Institute (IPN)
Mexico City, Mexico
{igor,gelbukh}@cic.ipn.mx

Abstract. The problem of automatic text segmentation is subcategorized into two different problems: thematic segmentation into rather large topically self-contained sections and splitting into paragraphs, i.e., lexico-grammatical segmentation of lower level. In this paper we consider the latter problem. We propose a method of reasonably splitting text into paragraph based on a text cohesion measure. Specifically, we propose a method of quantitative evaluation of text cohesion based on a large linguistic resource – a collocation network. At each step, our algorithm compares word occurrences in a text against a large DB of collocations and semantic links between words in the given natural language. The procedure consists in evaluation of the cohesion function, its smoothing, normalization, and comparing with a specially constructed threshold.

1 Introduction

In the recent decade, automatic text segmentation became a popular research area [4-13, 15, 17, 19]. In most cases, thematic segmentation is considered, i.e., the borders to be searched subdivide the text to rather long thematically self-contained parts. In contrast to most works in the area, in this paper we propose a method for a low-level, lexico-grammatical segmentation. The difference between these two segmentation tasks can be explained as follows.

A good application of thematic segmentation is automatic extraction of thematically relevant part(s) from a long unstructured file. When a file is too long for the user to read it through completely, a computer tool – a segmentation program – is quite handy. Another application of such segmentation is consulting a novice author on a better splitting his/her large and not yet brushed sci-tech text to balanced and thematically diverse sections.

As the main tool for thematic segmentation, the sets of terms belonging to each potential segment are considered. For example, the words most frequently used in the whole text are selected, the stop-words (mainly functional) are discarded, and then the similarity between adjacent potential segments is measured across the potential border as the cosine coefficient of occurrence numbers of the rest content words.

In such a task, the segmentation of the lower level, i.e., the division of text into sentences and paragraphs is supposed to have been done. Thus, paragraphs are considered as minimal text units with already determined lengths (measured in words or sentences) and terminological content.

However, this is by itself a problem faced by every author, namely, the problem of optimally splitting the text into paragraphs. One might consider rational splitting text

into paragraphs a component of general education at school in writing correct texts. However, numerous manuscripts of master-level students show that this component of school education is not efficient: many people who have to write sci-tech texts do not do it well. Specifically, grammatically correct texts written by humans are frequently subdivided into paragraphs in a rather arbitrary manner impeding smooth reading.

According to [22], the rational low level structuring of sci-tech texts is rather difficult even for humans. Besides splitting text into paragraphs it includes other difficult tasks, e.g., introduction of numbered or dotted items. In this paper we confine ourselves only to the task of splitting text into paragraphs.

It is a commonplace that singling out a paragraph conforms to some grammatical and logical rules that seem to be so far not formalized and thus not computable. From this point of view, the work [21] is an important step to this objective, but it supposes the problem of how to represent automatically by logical terms the meaning of any sentence and a text as a whole to have been solved, whereas modern computational linguistics only aims at this goal.

In this paper, we propose a method of lexico-grammatical segmentation of lower level, i.e., splitting texts into paragraphs. It is based on the following conjectures:

- Splitting text into paragraphs is determined by current text cohesion. Cohesive links are clustered within paragraphs, whereas the links between them are significantly weaker.
- At present, text cohesion has no formal definition. A human considers a text cohesive if it consistently narrates about selected entities (persons, things, relations, actions, processes, properties, etc.). At the level of semantic representation of text, cohesion is 'observable' in the form of linked terms and predicates of logical types, but it is not well explored how one can observe the same links 'at the surface.'
- In such conditions, it is worthwhile to suppose that text cohesion can be approximately determined through syntactic, pseudo-syntactic, and semantic links between words in a text.

  By pseudo-syntactic link we mean links that are similar to syntactic ones but hold between words of different sentences, for example, the link between *chief* and *demanded* in the text *She insulted her* ***chief****. He* ***demanded*** *on apology*.[1]
- Syntactic links are considered as in dependency grammars [14], which arrange words of any sentence in dependency trees. In the example (*she hurriedly*) ***went*** → *through* → (*the big*) ***forest***, the words out of parentheses constitute a dependency subtree (in this case, a chain) with the highlighted content words at the ends and the functional (auxiliary) word in between. The words within parentheses, as well as all other possible words of the sentence, are linked into the same tree, and other pairs of linked content words can be observed among them, such as ***hurriedly*** ← ***went*** or ***big*** ← ***forest***.

  Syntactic links between two content words are called collocations,[2] whereas functional words only subcategorize them. Indeed, collocations can be of various

[1] Formally, we define such a link to hold between words *a* and *b* (not necessarily in the same sentence) if in the text there is a word *c* coreferent with *a* and syntactically linked to *b*.

[2] There are different definitions of a collocation, e.g., [4, 11]. Some of them are based on statistical properties of word occurrences, e.g., mutual information. However, we define a collocation in the way explained here and use this term in this meaning throughout the paper.

classes: the first example above represents a combination of the ruling verb and its (prepositional) complement, while two other examples give combinations of verb or noun with their modifiers.

- Semantic links are well known. They connect synonyms, hyponym with a corresponding hyperonym, the whole with its part, or a word with its semantic derivative, like *possessor* to *possessive* or *to possess.* When occurring in the same text, such words are rarely linked syntactically. Their co-occurrences have other reason. Namely, the anaphoric (coreferential) entities can be represented in a text not only by direct repetitions and pronouns, but also by their synonyms or hyperonyms.
- A quantitative measure of cohesion implied by (pseudo-)syntactic and semantic links can be proposed. This measure experiences fluctuations along the texts, with maximums in the middle of the sentences and minimums between them. Some local minimums are deeper than others. Just they should be taken as splitting borders.

This paper proposes a method of quantitative evaluation of text cohesion. It compares word occurrences in a text against a large DB of collocations and semantic links in a given natural language. (Pseudo-)syntactic links are more important since within segments comparable with paragraphs by length no statistics of relevant terms can be collected. Taking into account the co-occurrences, our method processes cohesion function stage by stage, i.e., recurrently evaluates this function, smoothes it, normalizes, and compares it with a specially constructed threshold.

2 Databases of Collocations and Semantic Relations

An example of a huge DB containing semantic relations between English words is WordNet [3]. The EuroWordNet system [20] presents the same semantic relations for several other European languages. Regrettably, there are no collocations in these databases, in our definition of this term. Though semantic relations can be found in these sources, they alone do not solve the problem of evaluation of cohesion.

The only large DB of collocations and semantic links we know is CrossLexica system [1, 2]. Unfortunately, it covers only Russian language. However, we consider a system of this type as a base for our algorithm, in the hope that large resourced of this type will be available soon for other languages such as English or Spanish.

Let us discuss now the notion of pseudo-syntactic links in more detail, since such links are very important for our purposes.

Syntactic links hold within a sentence. Hence, if a pair of content words co-occurring in the same sentence is found in the collocation DB as potentially forming a link, the probability of this syntactic link between them in the given sentence is very high. Even if the link between the words in the text is different from the link registered in the DB for these words (e.g., the text contains *the woman who went...* while the DB contains *woman* ← *go*), the observed co-occurrence almost always gives evidence for some cohesion. In essence, we take into account anaphoric links in such cases.

Similarly, anaphoric links, which we cannot detect directly, permit us to suppose cohesion between lexemes *chief* and *demand* when they are registered in the DB as immediately linked but occurred in the adjacent (but different) sentences: *She insulted her* ***chief****. He* ***demanded*** *on apology*.

As to the semantic links, they hold across sentence borders even more frequently than the anaphorically conditioned pseudo-syntactic links mentioned above.

All these considerations give us grounds to ignore full stops in a text for detecting cohesive pairs in adjacent sentences.

3 Quantitative evaluation of text cohesion

In this section we present the algorithm of calculation of the text cohesion.

The algorithm uses a discrete variable i – the number of the word in the text. Punctuation marks have no numbers, and full stops ending sentences are not taken into account at this stage.

At each step, for the given position in the text the algorithm calculates the value of a special function that is to be compared with a threshold; the current position is then advanced. As soon as the value of the function crosses the threshold, a new paragraph is started, and the internal variables are reset. The details of the calculation of the function are explained below.

Let i be the current observation point within a text. To the left, some syntactically interconnected word pairs $\{p_k, q_k\}$, $p_k < q_k \le i$, have occurred. We define a partial measure of cohesion implied by k-th such pair as the function $U(q_k - p_k)$, where $q_k - p_k$ is the distance between words in the pair. Naturally, U decreases with the growth of $q_k - p_k$. We may suppose also that U depends on the class T_k of the syntactic relation within the pair and on the specific lexemes $\lambda(k)$ occurred at the points k. However, in a rough approximation we ignore the dependence on the class and the lexemes.

It is natural to suppose that the impact of the pair $\{p_k, q_k\}$ decreases at the point i along with its moving away from q_k. We evaluate this by the exponential factor $\exp(-\alpha(i - q_k))$. For the accumulated impact on the text cohesion of all (pseudo-)syntactically related words to the left of i (including pairs with the latter word in i), we have the following value:

$$\sum_{p_k \le i} U(q_k - p_k)\exp(-\alpha(i - q_k)). \tag{1}$$

For each semantically related pair $\{m_k, n_k\}$, $m_k < n_k \le i$, the partial measure of cohesion is taken as $V(n_k - m_k)$, and the dependence of V on the distance $n_k - m_k$ is generally different from that for V. Again, let us ignore the dependence of V on the semantic relation class S_k of the k-th pair and specific semantically linked lexemes. With the same exponent reflecting the 'forgetting' process, the measure of the total prehistory for semantically related pairs is:

$$\sum_{n_k \le i} V(n_k - m_k)\exp(-\alpha(i - n_k)). \tag{2}$$

By (1) and (2), the global cohesion function $F(i)$ satisfies the equation

$$F(i) = \exp(-\alpha)F(i-1) + Q(i), \tag{3}$$

where

$$Q(i) = \sum_{p_k = i} U(q_k - p_k) + \sum_{n_k = i} V(n_k - m_k). \tag{4}$$

The functions U and V were taken also in the exponential form:

$$U(n_k - m_k) = A \exp(-\beta (n_k - m_k)); \;\; V(n_k - m_k) = B \exp(-\delta (n_k - m_k)),$$

where A and B are constants with a ratio between to be selected experimentally; β = (1...3)/L; δ = (0.5...1)/L; L is the mean length of the sentence.

We can evaluate the equation (3) recurrently, since its current value is composed of the previous value taken with a coefficient less that 1 and the contribution $Q(i)$ of all pairs whose former points coincide with the current observation point.

Strictly speaking, the impact of the pairs extends backward to the very beginning of text, but really the only pairs distant not more than approximately $1/\alpha$ from the observation point are influent. It can be considered as the 'window width' of the computing algorithm.

4 Smoothing and Normalizing the Cohesion Function

The cohesion function $F(i)$ obtained above has two sources of randomness.

First, it is heavily saw-toothed, i.e., contains many local minimums and maximums, which that is caused by random scattering of content words in sentences. Before searching relevant minimums in this curve it is necessary to smooth it.

The simplest smoothing is linear [16], when the output (smoothed) function $G(i)$ is obtained by values of an input function $F(i)$ (to be smoothed) by the formula

$$G(i) = \sum_{j=0}^{\infty} F(i-j)R(j),$$

where $R(j)$ is reaction of the smoothing filter to a single value equal to 1. To conserve the scaling of the output function, we subject $R(j)$ to the normalizing condition:

$$\sum_{j=0}^{\infty} R(j) = 1.$$

The most convenient options for $R(j)$ are:

- Exponent $R(i) = (1-q)\, q^i$, where $i = 0, 1,...; 0 < q < 1$. This gives a recurrent formula:

  $$G(i) = qG(i-1) + (1-q)F(i).$$

  To effectively determine the result by few recent values of input function, q should be in the interval 0,5...0,7.
- Symmetric peak taking three adjacent values of the input function: $R(0) = R(2) = q/(1+2q)$; $R(1) = 1/(1+2q)$; $0 < q < 1$, so that

  $$G(i) = \frac{q}{1+2q} F(i-2) + \frac{1}{1+2q} F(i-1) + \frac{q}{1+2q} F(i).$$

  This option is not recurrent but also simple, since it stores only two previous values of F on each step. For $q = 1$, the three adjacent values of the input function are summed up with the equal weights 1/3 and the smoothing is the greatest, for $q = 0$ the smoothing is absent.

  The dispersion of independent input peaks decreases at the output

- by $(1-q)^2/(1-q^2)$ for the exponent (e.g., for $q = 0.5$ the decrease is 3);
- by $(1+2q^2)/(1+2q)^2$ for the peak (e.g., for $q = 0.5$ the decrease is 2.33).

At the same time, all slow components of $F(i)$ comes through the filter unimpeded.

The second source of the cohesion function randomness is the inevitable incompleteness of any DB of collocations. For any given language, in order to collect the collocations covering an average text up to, say, 95%, it is necessary to scan through (automatically, with further manual control and post-editing) such a huge and polythematic corpus of text that this needs too much labor. What is more, natural language is not static and new candidates for stable collocations appear in texts continuously.

In such a situation, it is more convenient to normalize the smoothed cohesion curve somehow. For this reason we propose to form the current mean value of the function $Q(i)$ given by the formula (4). The mean value is calculated through the whole document beginning from $i = 1$ by the recurrent formula

$$M(i) = (1 - \frac{1}{i})\, M\,(i-1) + \frac{1}{i} Q(i).$$

After passing several sentences, the current mean value experiences little fluctuation. Dividing $G(i)$ by $M(i)$, we obtain a normalized function that fluctuates respecting to 1, with local maximums near the middle points of sentences and comparable minimums at their ends. Besides of partial compensation of the DB incompleteness, the normalization decreases the arbitrariness of the selection of the functions U and V, especially with respect to the fertility of lexemes.

5 Splitting Text into Paragraphs

Now let us use the normalized cohesion function for splitting a text into paragraphs. We take into account the following considerations:

- The sequential point of splitting should be near a minimum of the normalized curve.
- The selected local minimum should be less than the recent minimums that have not been admitted paragraph boundaries at the previous steps of the algorithm.
- Usually an author unconsciously has in mind a mean length value P of a paragraph. If the distance from the current point i to the initial point j of the given paragraph is fairly less than P, the current cohesion measure is not so important, but near P, any noticeable minimum implies the decision to interrupt the paragraph.

These requirements are met by the continuous comparison of $G(i)/M(i)$ with the threshold

$$C(i,j) = C_0 + C_s((i-j)/(P+\Delta)^s,$$

where $C_0 \in [0.05 \ldots 0.2]$; $C_s = 1 - C_0$; $s \in [3\ldots5]$, $\Delta \in [1\ldots3]$.

As soon as $C(i,j)$ crosses $G(i)/M(i)$ in a point i_0, the word before the nearest full stop to the left of i_0 is taken as the end of the current paragraph, and the splitting algorithm continues scanning the text. The relations between functions $F(i)$, $M(i)$, $G(i)/M(i)$, and the threshold value $C(i,j)$ are illustrated in Figure 1.

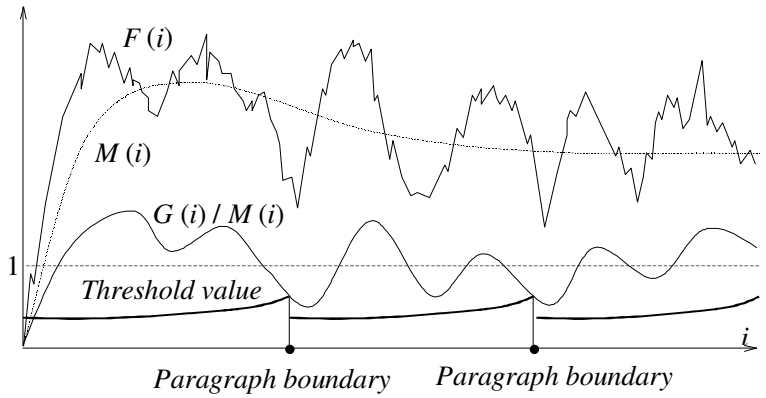

Figure 1. Correlation between cohesion-driven functions.

6 A simple experiment

For the simplest experimentation with the proposed algorithm, we have taken an article from Mexican newspaper with the following features: 997 words, 27 sentences, and 11 paragraphs selected by the author. All syntactic, pseudo-syntactic, and semantic links where marked manually beforehand.

The algorithm was applied to the text lacking paragraph boundaries, using the following parameters: $\alpha = 5 / L$, $\beta = 2 / L$, $\delta = 0.75 / L$, $A = B = 1$, $q = 0.6$, $C_0 = 0.1$, $s = 4$, $\Delta = 3$. The results were measured by *recall* and *precision* as compared to boundaries selected by the author.

Also we proposed the same task to three experts. The results of all experiments are gathered in the following table:

	Boundaries selected	*recall*	*precision*
Algorithm	9	0.60	0.66
Expert 1	9	0.50	0.66
Expert 2	6	0.50	0.83
Expert 3	13	0.80	0.61

One can see that the algorithm restores the paragraphs boundaries not worse than educated native speakers of Spanish. The results seem not persuasive but promising.

7 Conclusion and Future Work

A method of splitting text into paragraphs is proposed. It is based on the supposition that such splitting is implied by a measure of current text cohesion. The cohesion function is constructed basing on close co-occurrences of words pairs contained in a large database of collocations and semantic links. The computation includes several steps: estimation of the cohesion function, its smoothing, normalization, and compari-

son with a variable threshold depending on the expected paragraph length. Our preliminary experiments show promising results.

References

1. Bolshakov, I. A. *Multifunctional thesaurus for computerized preparation of Russian texts.* Automatic Documentation and Mathematical Linguistics. Allerton Press Inc. Vol. 28, No. 1, 1994, p. 13-28.
2. Bolshakov, I. A. *Multifunction thesaurus for Russian word processing*. Proc. of 4th Conf. on Applied Natural Language Processing, Stuttgart, 13-15 October, 1994, p. 200-202.
3. Fellbaum, Ch. (ed.) *WordNet as Electronic Lexical Database*. MIT Press, 1998.
4. Ferret, O. *How to Thematically Segment Texts by Using Lexical Cohesion?* Proc. of Coling-ACL-98, v. 2, 1998, p. 1481-1483.
5. Ferret, O., B. Grau, N. Masson. *Thematic segmentation of texts: two methods for two kinds of texts.* Proc. of Coling-ACL-1998, v. 1, p. 392-396.
6. Jobbins, A. C., L. J. Evett. *Text segmentation using reiteration and collocation.* In: Proc. of Coling-ACL-1998, v. 1, p. 614-618.
7. Hearst, A. M. *Multi-paragraph segmentation of expository text*. Proc. ACL-94. Las Cruces, N. M., USA, 1994, p. 9-16.
8. Hearst, A. M., C. Plaunt. *Subtopic Structuring for Full-Length Document Access*. Proc. ACM-SIGIR'93, 1993, p. 59-68.
9. Heinonen, O. *Optimal multiparagraph text segmentation by Dynamic Programming*. Proc. of Coling-ACL-98, v. 2, 1998, p. 1484-1486.
10. Litman, D., R.J. Passonneau. *Combining Multiple Knowledge Sources For Discourse Segmentation*. Proc. 31th Annual Meeting ACL Conference, 1993, Columbus, p. 108-115.
11. Kaufmann, S. *Second Order Cohesion*. Proc. PACLING'99 Conf., 1999, p. 209-222.
12. Kozima, H. *Text segmentation based on similarity between words.* Proc. of ACL-93, Columbus, Ohio, USA, 1993, p. 286-288.
13. Kurohashi, S., M. Nagao. *Automatic Detection of Discourse Structure By Checking Surface Information in Sentences*. Proc. Coling-94, Kyoto, 1994, p. 1123-1127.
14. Mel'cuk. I. *Dependency Syntax: Theory and Practice*. SUNY Press, NY. 1988.
15. Nomoto, T., Y. Nitta. *A Grammatico-Statistical Approach to Discourse Partitioning.* Proc. Coling-94, Kyoto, 1994, p. 1145-1150.
16. Oppenheim, A.V., R.V. Shafer. *Discrete-Time Signal Processing.* Prentice Hall. NJ, 1989.
17. Salton, G., A. Singhal, M. Mitra, C. Buckley.*Automatic Text Structuring and Summarization*. Information Processing & Management. V. 33(2), 1997, p. 193-207.
18. Smadja, F. *Retreiving Collocations from text: Xtract.* Computational Linguistics. Vol. 19, No. 1, 1993, p. 143-177.
19. Suzuki, Y. et al. *Segmentation and Event Detection of New Stories Using Term Weighting.* Proc. PACLING'99 Conf., 1999, p. 149-154.
20. Vossen, Piek (ed.). *EuroWordNet General Document*. Vers. 3 final. 2000, http://www.hum.uva.nl/~ewn.
21. Zadrozny, W., K. Jensen. *Semantic of Paragraphs*. Computational Linguistics. V. 17(2), 1991, p. 171-209.
22. Zobel, J. *Writing for computer science*. Springer. 1997.