Automatic Text and Data Stream Segmentation
Using Weighted Feature Extraction

Boris Dadachev
Cardiff School of Mathematics
Cardiff University
Cardiff CF24 4AG, UK
E-mail: DadachevBK@cardiff.ac.uk

Alexander Balinsky
Cardiff School of Mathematics
Cardiff University
Cardiff CF24 4AG, UK
E-mail: BalinskyA@cardiff.ac.uk

Helen Balinsky, FIMA
Hewlett-Packard Laboratories
Long Down Avenue, Stoke Gifford
Bristol BS34 8QZ, UK
E-mail: Helen.Balinsky@hp.com

George Forman
Hewlett-Packard Laboratories
10993 Fraser Rd. SE
Port Orchard, WA 98367, USA
E-mail: George.Forman@hp.com

Abstract
Automatic text and data stream segmentation is a fundamental NLP (Natural Language Processing) problem. It consists in breaking linearly a text into topically-consistent segments. Among other things, text segmentation can significantly improve the performance of various text mining algorithms and help individuals to get a quick grasp of long documents. In this article, a novel approach for automatic text and data stream segmentation is proposed and studied. The proposed automatic segmentation algorithm takes advantage of the Helmholtz-based feature extraction and unusual behavior detection. It is based on principles from image processing and especially on the Helmholtz Principle from the Gestalt Theory of human perception. It is entirely unsupervised and flexible to allow segmentation at different scales, such as short paragraphs and large sections. Initial results suggest it compares favourably with current state-of-the-art algorithms.

1. Introduction
Automatic text and data stream segmentation is a fundamental problem in text data mining. Even a moderately long document may consist of several relatively independent topics and parts. Such heterogeneity in documents can seriously affect the performance of classification and other mining algorithms. Applications range from screening of radio communication transcripts to document summarization, from automatic document classification to information visualization, from automatic filtering to security policy enforcement – all rely on, or can largely benefit from, automatic document segmentation. These are but a few examples of how text segmentation is finding its way into real-life applications.

In this article, a novel approach for automatic text and data stream segmentation is proposed and studied. Depending on the needs, a textual document could be required to be automatically partitioned into relatively small paragraphs or large sections. The proposed automatic segmentation algorithm takes advantage of the Helmholtz-based feature extraction and unusual behavior detection [1-3, 5]. It is based on principles from image processing and especially on the Helmholtz Principle from the Gestalt Theory of human perception. The main idea behind using the Gestalt Theory for text analysis is that humans actively use visual sensors when reading and writing documents. The human brain has evolved to work with images and as such, visual perception dominates our way of thinking. As mentioned in [6], the
Gestalt Theory is a single substantial scientific attempt to develop principles of visual reconstruction. “Gestalt” is a German word translatable as “whole”, “form”, “configuration” or “shape”. The first rigorous approach to quantify basic principles of Computer Vision is presented in [6].

The paper is organized as follows. Previous work is reviewed in Section 2. For the reader convenience, we briefly recall in Section 3 the Helmholtz Principle for textual and stream data; more detailed descriptions can be found in [1-3, 5]. In Section 4, we detail the steps of the proposed algorithm. Finally, conclusions are presented in Section 5.

2. Previous Work

The vast majority of segmentation algorithms rely on the same underlying idea: the quantification of lexical cohesion between different parts of a document. Lexical cohesion can be defined as “the cohesive effect achieved by the selection of vocabulary” [8] which intuitively means that changes in the vocabulary accompany topic shifts. A number of linguistic structures create lexical cohesion, such as word repetitions, synonyms and pronouns. However, those structures can be difficult to detect due to language ambiguities, often leading to the quantification of lexical cohesion solely based on some measure of word repetitions.

Text segmentation has been a large research area in the past twenty years and many different techniques have been proposed. A few algorithms, among the most significant ones, are briefly presented here. TextTiling [9] and C99 [4] compute the cosine similarity –using term frequencies– between, respectively, adjacent blocks of pseudosentences and all pairs of sentences in the document; boundaries are then placed in low similarity regions. Utiyama and Isahara [12], and Eisenstein and Barzilay [7], cast the segmentation problem into a probabilistic, Bayesian, framework and use it to determine the most likely segmentation.

It is worth mentioning that the performance of many algorithms can be improved using other features [7, 13], selected either manually or automatically. A common example of such features are cue phrases, like “Joining us…”, helping greatly for speech segmentation. However, relying on annotated corpora makes those techniques become both domain-dependent and more complex.

Our unsupervised segmentation algorithm makes use of the Helmholtz Principle to identify regions with low lexical cohesion. Let us first briefly explain the Helmholtz principle in human perception.

3. The Helmholtz Principle and Level of Meaningfulness

According to a basic principle of perception due to Helmholtz [6], an observed geometric structure is perceptually meaningful if it has a very low probability to appear in noise. As a common sense statement, this means that "events that could not happen by chance are immediately perceived". For example, a five-dot alignment exists in both images in Figure 1, but it can hardly be seen on the left-hand side image. Indeed, such a configuration is not exceptional in view of the total number of dots. On the contrary, in the right-hand side image, we immediately perceive the alignment as a large deviation from randomness that would be unlikely to happen by chance.

In the context of data mining, we shall define the Helmholtz principle as the statement that meaningful features and interesting events appear as large deviations from randomness. In the cases of textual, sequential or unstructured data, we derived a quantitative measure for such deviations in [1, 2]. This measure was inspired by [6], where similar measures were proposed for image processing.
We first pre-process the documents by splitting each sentence into words, using all non-alphabetical characters as delimiters. Then we down case all words and apply the Porter Stemming algorithm. Only stems of length at least two are considered, and sentences without such stems are removed. Stop-words removing may or may not be done; it is most of the time not necessary.

Let $\mathcal{P}$ denote a family of parts of the document $D$. Elements of $\mathcal{P}$ could be paragraphs, sections, pages of $D$ if the document has such logical units, or, more generally, several consecutive sentences. For any $P \in \mathcal{P}$, we can define a measure of meaningfulness of a word $w$ from $D$ inside $P$ as follows. If the word $w$ appears $m$ times in $P$ and $K$ times in the whole document $D$, then we first define the number of false-alarms $NFA(w, P)$ by the following expression (see [1] for details):

$$NFA(w, P) = \frac{1}{\binom{K}{m}} \cdot \frac{1}{\binom{K}{m}} \cdot \frac{1}{\binom{K}{m}} \cdot \frac{1}{\binom{K}{m}}$$

and the number $N$ is equal to $[L/B]$ where:
- $L$ is the length of the document $D$ and
- $B$ is the length of $P$ in the number of words.

Then we define a measure of meaningfulness of a word $w$ in $P$ as follows:

$$\text{Meaning}(w, P) := -\frac{1}{m} \log NFA(w, P).$$

The justification for using $\text{Meaning}(w, P)$ was given in [2] based on arguments from statistical physics. Many real-life documents do not have a natural subdivision into paragraphs or those may be unknown —like in the case of text segmentation—, in which cases we use several consecutive sentences (typically four or five consecutive sentences) as a generalized paragraph along with sliding windows.

4. Algorithm for Automatic Text and Data Stream Segmentation

Text documents typically contain several topics. Formally, our goal is to identify topically-coherent segments in a document of $d$ sentences; from the $d-1$ gaps between the document sentences, we are looking for the gaps which best separate the topics inside the document.

In most documents, topics have a hierarchical structure. Indeed, a book is divided into chapters, each chapter being divided into sections, themselves divided into paragraphs... Thus, the desired segment size will depend on the application and we believe a generic segmentation algorithm should not prefer a level of segmentation over another. This is why our algorithm takes a single input parameter: either the approximate number of segments desired or the approximate desired segment size (which are inverse numbers for a given text). This number
of segments defines the approximate segment size, which is used as the window size for Helmholtz feature extraction.

Let us now describe the main steps of our proposed algorithm for text segmentation.

4.1. Preprocessing and Initializations

Once a window size has been chosen for Helmholtz feature extraction, the document to segment is tokenized into sentences and words (see Section 3 for more preprocessing details).

Our algorithm quantifies lexical cohesion between consecutive sentences using gap scores, where a larger score indicates stronger cohesion. Those $d-1$ gap scores are initialized to 0.

4.2. Helmholtz scores computations

The algorithm uses sliding windows to compare local word frequencies in a window to the frequencies of the whole document, using the Helmholtz measure of meaningfulness (see Equation (1)).

More precisely, for each window, the Helmholtz scores of all words inside the window are computed; words with a strictly positive score are declared meaningful inside that window. It is important to note that some meaningful words may be present only in a small portion of the window. Such a word should only have an influence on the part of the window where it appears. We call it the activity stretch. Formally, the activity stretch of a word $w$ inside a window $W_i$ is the group of consecutive sentences between the first and last occurrence of $w$ in the sentences inside $W_i$. The gap scores in the current window are updated by adding the Helmholtz score, for each meaningful word in the window, to all gap scores encompassed by the word’s activity stretch. The exact algorithm is:

- For all possible sliding windows $W_i$:
  - For each word $w$ appearing in $W_i$:
    - Compute $\text{Meaning}(w, W_i)$ [according to Equation (1)]
    - If $\text{Meaning}(w, W_i) > 0$:
      - Determine the activity stretch of $w$ inside $W_i$.
      - Add $\text{Meaning}(w, W_i)$ to the gap score of each gap encompassed by the computed activity stretch.

Since there are more windows in the middle of a document than in its beginning or end, we use periodic boundary conditions, assuming the document sentences are arranged in a circular fashion.

Now, a small gap score indicates that a small number of meaningful words have their activity stretch encompassing the associated gap, and/or that those words are less meaningful (i.e., have smaller Helmholtz scores) than in other parts of the document. Thus, a small gap score indicates low lexical cohesion between sentences before and after the gap, making it a good candidate for placing a segment boundary. Figure 2 shows the typical shape of the gap scores curve for test documents. Sharp drops in the gap scores can be observed at the location of ground truth cuts.

4.3. Boundaries identification

The simplest approach to identify cuts (i.e., segment boundaries) is to look for the smallest gap scores. Unfortunately, gap scores are a highly variable function with a lot of local minima, even in vicinity of the global minimum. As it is common to proceed in image processing for similar problems, we use the diffusion equation $u_t = u_{xx}$ (or the one-dimensional scale-space theory) to smooth the gap-score curve, and find stable and reliable critical points.
Let us recall the main property of the one-dimensional scale-space theory: increasing the smoothing parameter (or diffusion time) results in the decrease of the number of critical points [10], as desired. Note that this property is valid only in one dimension and not in any other dimensions.

From the numerical point of view, the evolution under the diffusion equation is just a convolution of the gap-score curve with a Gaussian filter $\frac{1}{\sqrt{2\pi t}} e^{-x^2/2t}$. Using the three-sigma rule, we convolve the filter only in a neighborhood of size $3t$ around every gap. We increase the parameter $t$ with small steps until the number of minima reaches the desired number of cuts. Smoothed gap scores are represented in Figure 3 by the red curve.

It is a well-known property of the Brownian motion that, under the diffusion process applied in the previous step, each point shifts on average by a distance less than or equal to $t$. We are only interested in the locations of the critical points but one cannot predict the exact motion of a given point. Some points will shift by a small distance only from their original location, while some others will go much farther (i.e., at a distance greater than $t$ from their original location). In our algorithm, we consider that a critical point can shift at a distance up to $1.5t$.

Thus, we find the “true” local minima in the original noisy curve, in the $1.5t$ vicinity of the shifted critical points (i.e., the local minima computed from the smoothed curve) and output those “true” local minima as the segment boundaries.

5. Evaluation

Our algorithm is tested against the C99 algorithm [4] that is traditionally used for comparison (the author’s implementation is publicly available). Evaluation documents are typically obtained by combining together ten randomly selected documents (of related content) of varying size. We use Choi’s dataset, made of 700 documents with segment sizes ranging from 3 to 11 sentences [4]. We also create a more realistic dataset with bigger segments (from 30 to 50 sentences) using the same methodology. The evaluation metric we use is WindowDiff [11]. Numerical results are presented in Table 1. Note that smaller scores indicate better performance and that the number of segments in each document is considered known.

For short documents, C99 produces slightly better results. However, our algorithm outperforms C99 on longer, more realistic, documents. This behaviour is not surprising given the statistical nature of the Helmholtz principle. Another important observation is that C99 is extremely sensitive to stop-words (abbreviated as “s-w” in Table 1). In contrast, our algorithm is robust and performs even better without stop-words filtering, making it readily available for
different domains and jargons: indeed, there is no universal list of stop-words. This also seems to indicate that stop-words contribute to the lexical cohesion effect.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.122</td>
<td>0.138</td>
<td>0.100</td>
<td>0.086</td>
<td>0.154</td>
</tr>
<tr>
<td>C99 (s-w not removed)</td>
<td>0.317</td>
<td>0.378</td>
<td>0.358</td>
<td>0.393</td>
<td>0.457</td>
</tr>
<tr>
<td>Helmholtz (s-w removed)</td>
<td>0.164</td>
<td>0.178</td>
<td>0.181</td>
<td>0.145</td>
<td>0.148</td>
</tr>
<tr>
<td>Helmholtz (s-w not removed)</td>
<td>0.160</td>
<td>0.181</td>
<td>0.162</td>
<td>0.138</td>
<td>0.101</td>
</tr>
</tbody>
</table>

TABLE 1. Performance comparison of C99 and Helmholtz algorithms (using WindowDiff)

6. Conclusions and Future Work

Automatic document segmentation is an important technique that can be used to improve the performance of many Natural Language Processing tasks. In this article, we introduced a new segmentation algorithm for text and data streams based the Helmholtz Principle. It takes a single intuitive parameter: the approximate number of segments desired. This makes our algorithm flexible and usable at different scales, ranging from small paragraphs to large sections such as book chapters.

The standard evaluation datasets are made from concatenated documents to have non-ambiguous ground truth segments. This is necessary for objective comparisons of algorithms; however this is unlikely to match real-world applications. As part of the next steps, we would like to perform more substantial testing on different types of documents and corpora.

REFERENCES