Optional ERIC Coversheet — Only for Use with U.S. Department of Education Grantee Submissions

This coversheet should be completed by grantees and added to the PDF of your submission if the information required in this form **is not included on the PDF to be submitted**.

INSTRUCTIONS

- Before beginning submission process, download this PDF coversheet if you will need to provide information not on the PDF.
- Fill in all fields—information in this form **must match** the information on the submitted PDF and add missing information.
- Attach completed coversheet to the PDF you will upload to ERIC [use Adobe Acrobat or other program to combine PDF files]—do not upload the coversheet as a separate document.
- Begin completing submission form at https://eric.ed.gov/submit/ and upload the full-text PDF with attached coversheet when indicated. Your full-text PDF will display in ERIC after the 12-month embargo period.

GRANTEE SUBMISSION REQUIRED FIELDS

Title of article, paper, or other content

Automated Paragraph Detection Using Cohesion Network Analysis

All author name(s) and affiliations on PDF. If more than 6 names, ERIC will complete the list from the submitted PDF.

Last Name, First Name	Academic/Organizational Affiliation	ORCID ID
Botarleanu, Robert-Mihai	University Politehnica of Bucharest	
Dascalu, Mihai	Academy of Romanian Scientists	
Crossley, Scott	Georgia State University	
McNamara, Danielle	Arizona State University	

Publication/Completion Date—(if *In Press,* enter year accepted or completed) September 28, 2022

Check type of content being submitted and complete one of the following in the box below:

- ☐ If article: Name of journal, volume, and issue number if available
- ☑ If paper: Name of conference, date of conference, and place of conference
- ☐ If book chapter: Title of book, page range, publisher name and location
- ☐ If book: Publisher name and location
- ☐ If dissertation: Name of institution, type of degree, and department granting degree

Conference on Smart Learning Ecosystems and Regional Development, September 28, 2022, Bucharest, Roma

DOI or URL to published work (if available) doi.org/10.1007/978-981-19-5240-1_5

Acknowledgement of Funding— Grantees should check with their grant officer for the preferred wording to acknowledge funding. If the grant officer does not have a preference, grantees can use this suggested wording (adjust wording if multiple grants are to be acknowledged). Fill in Department of Education funding office, grant number, and name of grant recipient institution or organization.

"This work was supported by U.S. Department of Education **[Office name]** Institute of Education Scien through **[Grant number]** R305A1801 to **Institution]** Arizona State U. The opinions expressed are those of the authors and do not represent views of the **[Office name]** Institute of Education Scie or the U.S. Department of Education.

Automated Paragraph Detection Using Cohesion Network Analysis

Chapter · September 2022

DOI: 10.1007/978-981-19-5240-1_5

4 authors, including:

Mihai Dascalu
Polytechnic University of Bucharest
298 PUBLICATIONS **2,306** CITATIONS

SEE PROFILE

Danielle McNamara
Arizona State University
482 PUBLICATIONS **21,296** CITATIONS

SEE PROFILE

Automated Paragraph Detection using Cohesion Network Analysis

Robert-Mihai Botarleanu[1], Mihai Dascalu[1,2],
Scott Andrew Crossley[3], Danielle S. McNamara[4]

[1] University Politehnica of Bucharest, 313 Splaiul Independentei, 060042, Bucharest, Romania
robert.botarleanu@stud.acs.pub.ro, mihai.dascalu@cs.pub.ro
[2] Academy of Romanian Scientists, Str. Ilfov, Nr. 3, 050044, Bucharest, Romania
[3] Georgia State University, Department of Applied Linguistics/ESL, Atlanta, GA 30303, USA
scrossley@gsu.edu
[4] Arizona State University, Department of Psychology, PO Box 871104, Tempe, AZ 85287
dsmcnama@asu.edu

Abstract. The ability to express yourself concisely and coherently is a crucial skill, both for academic purposes and professional careers. An important aspect to consider in writing is an adequate segmentation of ideas, which in turn requires a proper understanding of where to place paragraph breaks. However, these decisions are often performed intuitively, with little systematicity in sequencing ideas. Thus, an automated method of detecting the optimal hierarchical structure of texts using quantifiable features could be a valuable tool for learners. Here, we aim to define a framework grounded in Cohesion Network Analysis to establish the structure of a text by modeling paragraphs as clusters of sentences. The analogy to clustering enables us to identify paragraph breaks that maximize inter-paragraph separation while ensuring high intra-paragraph cohesion. Our approach consists of two steps acted on texts without paragraph breaks. First, the number of paragraphs is automatically inferred with an absolute error of 1.02 using a Recurrent Neural Network, which relies on text features and cohesion flow. Second, paragraph splits are detected using two algorithms: *top k* which selects the largest cohesion gaps between adjacent utterances, and *divisive clustering* which iteratively splits the text into paragraphs. Silhouette scores are used to assess performance and the obtained values denote adequately inferred structures.

Keywords: Cohesion Network Analysis, Paragraph Marking, Clustering, Sentence Embeddings.

1 Introduction

Learning to write is an important aspect of education and a useful skill across many circumstances. An important aspect of writing is text structure, which needs to be taken into account to convey content and facilitate understanding. Stark [1] found that paragraphs are discourse units that affect what ideas are considered to be important. As such, paragraph breaks represent a central delimitator of ideas and impact the structure

and the coherence of a text. However, the task of identifying where to place paragraph breaks in a sequence of sentences without them is not trivial. This challenge is mainly because paragraph composition relates to the flow and sequencing of ideas, in tight relation to text cohesion, both at local (i.e., in-between sentences) and global (i.e., among paragraphs) levels. In addition, writers have a personal style and may place paragraphs differently. For instance, they may group more sentences to maximize the content of each paragraph, or they may prefer to use a more fractionated structure with more fine-grained, individualized ideas per paragraph. In more extreme cases, students may have single paragraphs with many sentences or a large number of very short paragraphs, with only one sentence each.

Our proposed system analyzes the quality of a text's paragraph structure and enables automated feedback integrated into a smart learning environment, to provide instruction to students on how to better organize sentences into cohesive paragraphs. The model can be considered as groundwork for analyzing texts on a topological level, to guide learners towards improving their writing, in particular text structure and organization, by providing easily understandable, explicit feedback.

In this paper, we present a two-stage approach to the issue of paragraph identification. First, we detect the optimal number of paragraphs through a Recurrent Neural Network. Second, we identify the optimal structure of paragraphs in the text based on text cohesion, to maximize inter-paragraph separation, while ensuring high intra-paragraph cohesion. Thus, our research question is the following: to what extent is our automated model capable to predict the optimal number of paragraphs in a text, as well as how adequate is the proposed topology of a text derived from the proposed paragraph segmentation?

2 Related Work

Paragraph detection is a subtask of the wider concept of paragraph segmentation. It is a separate topic from the detection of already existing paragraphs, such as those that can be found in PDF files [2]. The task of detecting the optimal paragraph structure of a text in the current study falls under Automated Writing Evaluation (AWE) [3] because our tool suggests paragraph break positions in student writings in order to maximize the readability of their work while structuring and compartmentalizing sentences into distinct, cohesive groups. In contrast to Automated Essay Scoring (AES), AWE systems provide targeted feedback to users to help them improve their texts. The problem of identifying the optimal paragraphs is either modeled as: a) a text with all paragraphs removed in which boundaries need to be established or b) a sequence of sentences for which a paragraph break exists, and its optimal location needs to be identified.

The problem of detecting paragraph boundaries was proposed by Genzel [4] for use in downstream tasks, such as grammar checking or as a restoration step after OCR processing. The arbitrary nature of paragraph boundary placement is noted. In this work, paragraph detection is modeled as a classifying task that establishes whether a sentence is paragraph-starting or not. The model used is a sparse voted perceptron, with an emphasis being placed on the features the model learns to use, which may be indicative of

what constitutes a paragraph-starting sentence for humans. Various statistical features were used, such as the cosine similarities between sentences or their first words, parse tree statistics, part of speech statistics, surface statistics (e.g., length of sentences), centering types, and others. Accuracies between 64% and 82% were reported for a model trained on "War and Peace" by Leo Tolstoy and evaluated on other texts. The main issue with this approach is that there is a significant imbalance between paragraph-starting and non-paragraph-starting sentences, which means that: a) the model can outperform a baseline that always predicts the majority class by 17 percentage points for some texts, while b) the model is 6 percentage points below the oracle baseline for other texts.

Another example of paragraph detection can be found in Sporleder and Lapata [5], who introduced a state-of-the-art system for marking paragraphs by proposing methods of paragraph detection using hand-crafted features. They achieved a performance that is within 6% of the human baseline, with an accuracy of 82.91% on their corpus. The authors used fiction texts, news, and parliamentary transcripts in three languages (English, Greek, and German) and included the following features:

- Sentence signatures composed of part-of-speech tags;
- Parse tree complexity considering the distribution of parts of speech and other geometrical features, such as the branching factor and depth;
- Statistical features such as the distance between the current sentence and the previous paragraph break, the length of the sentence, and the relative position in the text;
- Other heuristics such as the presence of quotation and punctuation marks;
- Word features, for the first three words of a sentence and the aggregated words of a sentence.

However, their method of modeling the task of paragraph detection as a classification problem has several drawbacks, similar to Genzel [4]. First, the number of paragraphs in a text is significantly lower than the number of sentences. This means that, given any succession of sentences, there is a bias towards the sentences belonging to the same paragraph which produces an imbalanced classification dataset. As such, measuring accuracy may be misleading. Genzel [4] also reported the results of an oracle that only predicts the majority class. Second, the introduced features do not take into account the overall structure of the text and consider only surface indicators of sentence complexity and length to split paragraphs. Because the primary purpose of a paragraph is to structurally group cohesive blocks of sentences, we hypothesized that a graph-like cohesion-centered structure of a text would capture the properties of paragraphs more effectively.

The purpose of our work is to detect adequate locations in a text where a writer should consider placing paragraph breaks that maximize two properties: intra-paragraph cohesion and decoupling between paragraphs. As such, we propose an unsupervised approach that takes into account the structural information extracted from a text. We assume that sentences in the same paragraph should have a higher cohesion with each other, as compared to sentences in differing paragraphs. This is similar to how points within the same clusters should be closer to each other than they are to points from other clusters.

Therefore, we model a text in a clustering space, where sentences represent nodes, paragraphs are clusters, and text cohesion is used to measure the relatedness of nodes. In other words, we analyze how a document is split into paragraphs by measuring whether the clusters (i.e., the paragraphs) form distinct sequences of points (i.e., sentences) that are separable. With this in mind, a text can be viewed as having a good paragraph structure if each paragraph is highly cohesive and, simultaneously, if there exists a clear separation between one paragraph and another.

Our method is grounded in Cohesion Network Analysis [6] which considers text structure based on semantic links established between different constituents (i.e., sentences, paragraphs, and the entire document). Our approach generates new paragraph structures for texts by maximizing the two previously mentioned properties (i.e., intra-paragraph cohesion and inter-paragraph decoupling). We propose a two-step approach. First, we extract various information from the text to predict the optimal number of paragraphs. Second, we develop two algorithms to detect the optimal configuration of the sentences into paragraphs. We compare the structural clustering metric scores from our two models to those measured in the original, human-paragraphed texts.

3 Method

3.1 Corpus

We used a combination of documents from the TASA (Touchstone Applied Science Associates, Inc.) corpus (http://lsa.colorado.edu/spaces.html) and essays gathered using the Writing Pal intelligent tutoring system (ITS) [7] in various experiments. The TASA corpus was selected because the initial texts were split into different self-defined short documents (1-7 paragraphs) by experts, whereas the human essays represent a more relaxed and free-form structure, which varies greatly due to a large number of writers. From these, only texts that have at least 3 and at most 7 paragraphs were selected, resulting in a dataset containing 4,704 filtered texts of which 3,893 are taken from the TASA corpus and 811 are essays written by students (see Fig. 1 for the histogram with the distribution of paragraph length).

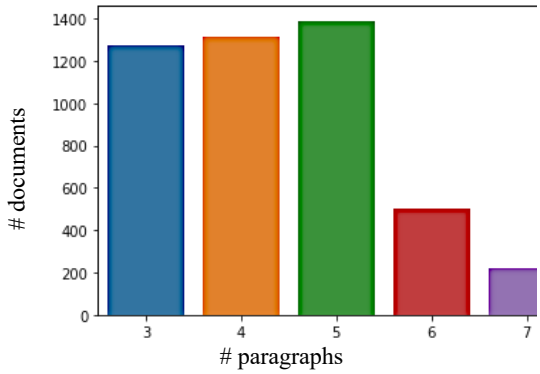


Fig. 1. Histogram of paragraph counts for the paragraph detection corpus.

3.2 Predicting the number of paragraphs

First, we built a model to predict the number of paragraphs in an unstructured text. A list of features from each text is generated to train a regressor. These features correspond both to surface-level statistics, as well as features derived from the cohesion between sentences, computed using cosine similarity on sentence embeddings. The features extracted included:

- Counts of words, sentences, different word lemmas, and stopwords;
- Distributions of part of speech tags using the Penn Treebank Part-of-speech tagger and determining the frequency of each tag per document;
- Counts of specific connectors (e.g., cause and effect, comparison, emphasis, etc.); in total, 88 connectors are used, representing common sentence and paragraph boundaries taken from various word lists;
- A measure of cohesion flow operationalized as a vector with cohesion scores between two consecutive sentences; we use the cosine similarity between two sentences to measure their semantic distance and, through the sequence of such distances in the text, the cohesiveness of the entire text;s
- Various statistical measures were applied to the cohesion flow: mean, standard deviation, kurtosis, skewness, 0.1/0.25/0.5/0.75/0.9 quantiles, 25/50/75 percentile values.

Two types of models are considered while assessing the flow of cohesion, namely: a) a pre-trained word2vec [8] model using the Google News word embeddings, and b) a word2vec model trained on the Corpus of Contemporary American English – COCA [9] dataset.

The aggregation of individual word embeddings into phrase-level vector representations is achieved either through an unweighted mean of word vectors or through the Smooth Inverse Frequency (SIF) [10] which assigns a larger weight to less common words:

$$v_s = \frac{1}{|s|}\sum_{w \in s} \frac{a}{a+p(w)} v_w \quad (1)$$

where v describes an embedding vector, s is a sentence, w is a word in s, p is the probability of seeing w in s, and v_w is the vector representation of w.

The regressor used for the task of predicting the number of paragraphs in a text (see Fig. 2) is a neural network composed of a hidden linear layer with 64 neurons, followed by a Rectified Linear Unit activation [11] for the statistical input features, and a single LSTM [12] layer with 32 units that receives the cohesion flow vector as input. The results of these two layers are concatenated, and the final prediction is given through an output layer with a single neuron.

The intuition behind the architecture introduced in Fig. 2 is that the fully connected layer should generate new representations of the statistical features, while the LSTM should learn to understand that cohesion gaps are correlated with the number of paragraphs. The model is trained using the Adam optimizer [13] for 10 epochs and optimizes the mean squared error. A cosine annealing learning rate scheduler [14] is used,

with the learning rate going from a maximum of 0.01 to a minimum of 0.001. Additionally, we perform a z-score normalization of the target variable and report the results after denormalization.

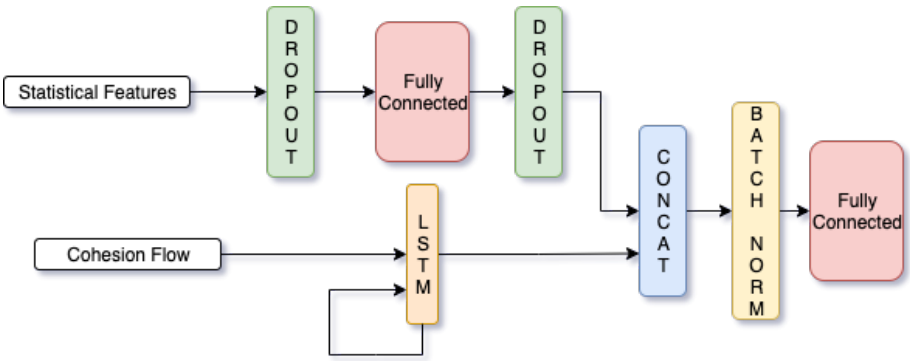


Fig. 2. Architecture for the paragraph counting model.

3.3 Identifying paragraph breaks

Once the number of paragraphs has been estimated, we can find the optimal paragraph structure of the text. Two algorithms – *top k* and *divisive clustering* – were tested to detect where paragraphs should be placed in an arbitrary set of sentences. Each algorithm assumes that the number of paragraphs (P) is given as input. Both algorithms preserve the original order of the sentences in the text. The *top k* algorithm detects the highest P cohesion breaks and marks them as paragraph splits. A cohesion break is defined as a succession of two adjacent sentences for which the cohesion between them is low. The *divisive clustering* considers the following:

a. Consider all sentences to belong to one cluster.
b. For (P – 1) repetitions:
   (1) Find the sentence that is most dissimilar (on average) to its cluster.
   (2) Split the paragraph such that the sentence found previously and all sentences following it in the cluster are assigned to a new paragraph.

3.4 Evaluation

Using the analogy that representations of sentences occur in a clustering space wherein the paragraphs are clusters, we adapt the Silhouette score [15] to assess the performance of our two algorithms. The Silhouette score is a frequently employed clustering metric used to evaluate the extent to which an algorithm maximizes intra-cluster similarity and minimizes inter-cluster similarity. We expect good paragraphs to lead to high Silhouette scores since sentences inside such a paragraph are likely to be more connected than those belonging to different paragraphs.

Our adapted Silhouette score considers the following steps:

1. Compute the average distance between each point *i* and all other points in the clustering space (C_j denotes sentences from cluster/paragraph C_j):

$$a(i) = \frac{1}{|C_i|-1}\sum_{j \in C_i, i \neq j}(1 - cohesion(i,j)) \quad (2)$$

2. For each point *i*, compute the distance to the nearest neighbor in another cluster:

$$b(i) = \min_{k \neq i} \frac{1}{|C_k|}\sum_{j \in C_k}(1 - cohesion(i,j)) \quad (3)$$

3. Compute the Silhouette score for each point *i* as:

$$s(i) = \frac{b(i)-a(i)}{\max(a(i),b(i))} \quad (4)$$

4. Finally, the document silhouette score is the average of the individual silhouette scores for all sentences (*D* denotes the entire document):

$$S = \frac{1}{|D|}\sum_{i \in D} s(i) \quad (5)$$

As such, the Silhouette score provides higher values to texts where sentences are well placed inside their paragraphs.

4 Results

Our corpus was split into a training and a test set with a ratio of 4:1 for each sub-task. The first component consists of a regressor that predicts the number of paragraphs in a text. Results from Table 1 indicate that neither the word2vec corpus nor the aggregation method (i.e., unweighted *mean* of the word vectors versus Smooth Inverse Frequency) affected the performance of the four models. The mean absolute error measures the average difference between the actual and the predicted number of paragraphs. An MAE of 1 reflects a difference of 1 paragraph in the proposed text structure, making our model usable in practice.

Table 1. Mean Absolute Error for the paragraph counting regressor as a function of aggregation method and word2vec corpus.

Aggregation Method	word2vec Corpus	Mean Absolute Error (MAE)
Mean	Google news	1.03
SIF	Google news	1.02
Mean	COCA	1.03
SIF	COCA	1.02

Table 2 provides the Silhouette scores measured using all variations of aggregation methods, word2vec models, and the two proposed algorithms (i.e., *top k* and *divisive*

clustering). We considered *P* to be the predicted number of paragraphs resulting from the previous sub-task, as well as the actual number of paragraphs. The later assessment using the actual number of paragraphs from the reference tests was performed to evaluate this sub-task individually, without errors induced by the previous prediction component. Results indicate that the use of the word2vec embeddings trained on the Google News corpus led to better performance in detecting optimal paragraphs. This is consistent with the idea that word embeddings trained on larger corpora are better suited for measuring the semantic relatedness between sentences because they are more likely to capture the relations between words. In addition, divisive clustering outperforms the simple "top k" algorithm, showing that a top-down approach is better suited for this task. Results also indicate there is no noticeable difference between using an unweighted average of the word vectors of a sentence and using a weighted average where the weights are given by the frequency of a word's apparition (i.e., using SIF).

Table 2. Silhouette scores for the paragraph detection test set.

Aggregation Method	word2vec Corpus	Paragraph Detection Algorithm	Silhouette score using the predicted # of paragraphs	Silhouette score using the actual # of paragraphs
Mean	Google news	Top k	0.22	0.19
SIF	Google news	Top k	0.22	0.19
Mean	COCA	Top k	0.19	0.15
SIF	COCA	Top k	0.19	0.16
Mean	**Google news**	**Divisive clustering**	**0.23**	**0.20**
SIF	**Google news**	**Divisive clustering**	**0.23**	**0.20**
Mean	COCA	Divisive clustering	0.20	0.17
SIF	COCA	Divisive clustering	0.21	0.17

An additional experiment was performed that relied only on the Silhouette score as the selection criterion to determine the optimum number of paragraphs. This was achieved by running the paragraph detection algorithms on *all* paragraph counts from 3 to 7, followed by the evaluation of corresponding Silhouette scores. This experiment provided insights into whether the Silhouette score is a metric correlated to the number of paragraphs in the text and whether the paragraph count regressor is necessary for the algorithm to detect the correct number of paragraphs (see Table 3).

The mean average errors obtained in this experiment are approximately twice as high as those measured using the regressor (see MAE values in Table 1). This indicates that having a separate model to estimate the correct number of paragraphs is, indeed, useful as cohesion is a key constituent, but not sufficient by itself, to accurately predict how many paragraphs a text should contain.

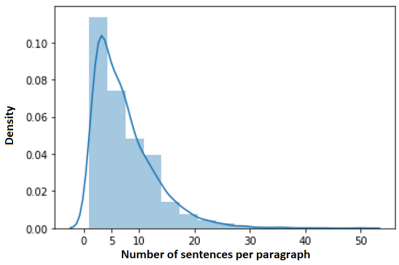
Table 3. Mean Absolute Error using the Silhouette score as an indicator of the number of paragraphs.

Aggregation Method	word2vec Corpus	Paragraph Detection Algorithm	Mean Absolute Error (MAE)
Mean	Google news	Top k	2.52
SIF	Google news	Top k	2.53
Mean	COCA	Top k	2.45
SIF	COCA	Top k	2.48
Mean	Google news	Divisive clustering	2.51
SIF	Google news	Divisive clustering	2.53
Mean	**COCA**	**Divisive clustering**	**2.39**
SIF	COCA	Divisive clustering	2.42

5 Discussion

This study introduces an automated method to detecting the optimal hierarchical structure of texts using quantifiable features. The framework is grounded in Cohesion Network Analysis and models paragraphs as clusters of sentences, allowing us to identify paragraph breaks that maximize inter-paragraph separation while ensuring high intra-paragraph cohesion. Fig. 3 describes how the paragraphs in the generated texts are structured as a function of the number of sentences. In the original texts, the average number of sentences per paragraph was 4.21, with the 50th percentile being at 4 sentences per paragraph. The generated paragraphs have an average of 7.4 sentences, with the 50th percentile being at 6 sentences per paragraph. Thus, our system appears to generate longer paragraphs than those found in the original texts, which would suggest that higher Silhouette scores result from more compacted blocks of sentences.

(a)



(b)

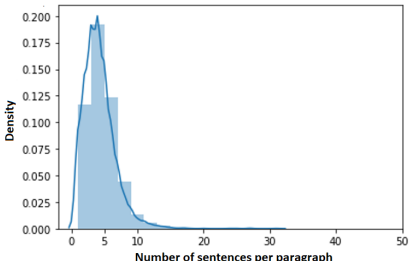


Fig. 3. Distribution plot showing the number of sentences in each (a) original paragraph and (b) generated paragraph.

To better understand how paragraph breaks are marked, we sample from each of the three possible cases in which the splits into paragraphs are different than the initial ones (see Table 4). In the first example, an intermediate paragraph is generated from the second and third source text paragraphs. While the beginning of this supplementary paragraph is not ideal since it starts with “Instead of like a pack of wolves” which requires the previous sentence for continuity, the final paragraph in the altered text more clearly delimits the essay conclusions. In the second example where an equal number of paragraphs is used, the generated text in our approach elects to start the second and third paragraphs with connectors such as “Although” and “For example”. In the final example where only one paragraph is generated using all three paragraphs from the source text, our system detects that the three paragraphs are highly interconnected and can be more fluently expressed as a single paragraph.

Overall, we find that our method generates more paragraphs than in the original text; the system generates an equal number of paragraphs to the original text, but at different places; and the system generates a greater number of paragraphs than in the original text. Given that the paragraph count regressor has an MAE of 1.24, deviations of more than one paragraph are rare. The results indicate that the introduced paragraph breaks make quite good sense as the system splits paragraphs based on the similarity between ideas. For example, the new collided version of the sentences from the last sample into one paragraph seems more adequate and cohesive, especially in contrast to the highly segmented initial version which had two paragraphs with only one sentence each.

6 Conclusions

A novel automated approach was introduced to analyze how texts are structured into paragraphs by considering cohesive clusters of sentences. Cohesion Network Analysis was used to capture text structure and maximize the cohesion between sentences in a paragraph while keeping distinct paragraphs separate.

The marking of paragraphs is performed as a two-stage process. First, a regressor using a Recurrent Neural Network is trained to estimate the number of paragraphs in a text; this sub-task achieves a mean absolute error of 1.02, which indicates that we can predict the number of paragraphs in our combined corpus with an average error of around 1 paragraph. Second, the Silhouette clustering score, in conjunction with a paragraph splitting algorithm, is used to determine an optimal paragraph structure for the predicted number of paragraphs.

Our primary contribution is that our proposed clustering approach is unsupervised and independent of the quality of the human paragraph segmentations within a corpus. Our method does not require training and it offers an explainable and intuitive approach to find the optimal paragraph structure of a text. Moreover, our method provides a quantitative measure of a text’ paragraph topology.

Table 4. Samples of generated paragraphs.

Actual Paragraphs	Generated Paragraphs
More generated paragraphs	
I believe that any kind of material such as books, movies, magazines, etc., should be censored except for mature content such as pornography, violent and horrid movies and video games. But that's why we have ORGANIZATION1's made. So when asked, we can show our age. Some age groups react differently to mature contents. Usually NUM1 and up is a tolerable level for maturity to set in. Maturity is maintained at different paces in all age groups. For example, perhaps a NUM2 year old is more mature than an NUM3 year old. It is when you as a individual can act like a human being. Instead of like a pack of rabid wolves. Just because you do not act mature dosen'CAPS2 mean you don't know how to be civilized. The best way to prevent a child from discover something inappropriate or vulgar is to, separate the books by an age grouping system. Keep adult videos out of site and reach of children. Also put safety locks on your CAPS2.V. to prevent kids from sneaking behind your back	I believe that any kind of material such as books, movies, magazines, etc., should be censored except for mature content such as pornography, violent and horrid movies and video games. But that's why we have ORGANIZATION1's made. So when asked, we can show our age. Some age groups react differently to mature contents. Usually NUM1 and up is a tolerable level for maturity to set in. Maturity is maintained at different paces in all age groups. For example, perhaps a NUM2 year old is more mature than an NUM3 year old. It is when you as a individual can act like a human being. Instead of like a pack of rabid wolves. Just because you do not act mature dosen'CAPS2 mean you don't know how to be civilized. The best way to prevent a child from discover something inappropriate or vulgar is to, separate the books by an age grouping system. Keep adult videos out of site and reach of children. Also put safety locks on your CAPS2.V. to prevent kids from sneaking behind your back
Equal number of paragraphs	
There are some ways to be original but most have already been done. Some will use others originals and maybe perfect it . Although we value uniqueness and originality, also we like to see others enhance someone else idea to make it better. There is a ongoing debate about can people be truly original. I think in some ways this is true because people can still have ideas that no one has thought of before. Although, people use others ideas they can make them better in the future or make them look like they are themselves original. For example, people buy clothing from the store and the clothing is a designer's creation. People take that and put it with other designer clothing and make their own original style. I think that is kind of original, because its their own way of wearing the clothing they buy. So although people may use others design or something similar to it. It can still be original it all depends on how people use it in a different way.	There are some ways to be original but most have already been done. Some will use others originals and maybe perfect it. Although we value uniqueness and originality, also we like to see others enhance someone else idea to make it better. There is a ongoing debate about can people be truly original. I think in some ways this is true because people can still have ideas that no one has thought of before. Although, people use others ideas they can make them better in the future or make them look like they are themselves original. For example, people buy clothing from the store and the clothing is a designer's creation. People take that and put it with other designer clothing and make their own original style. I think that is kind of original, because its their own way of wearing the clothing they buy. So although people may use others design or something similar to it. It can still be original it all depends on how people use it in a different way.
Fewer generated paragraphs	
I do think that there should be a censorship in not just in libraries, but everywhere. Personally, I think that the way that the libraries have the books are appropriate and if the parents do not want their children going anywhere that is not privy to them keep a hand length away. As for the parents, the parents know the areas that interest them, therefore the parents should go there	I do think that there should be a censorship in not just in libraries, but everywhere. Personally, I think that the way that the libraries have the books are appropriate and if the parents do not want their children going anywhere that is not privy to them keep a hand length away. As for the parents, the parents know the areas that interest them, therefor the parents should go there

We demonstrate that our method works best in a two-stage approach, wherein we first determine the number of paragraphs that should exist in the given text, and then select the appropriate sentence distribution for those paragraphs. Our experiments have shown that attempting to simultaneously determine a) the number of paragraphs and b) what sentences belong to them, leads to inferior quantitative outcomes. Our system can generate paragraphs similar to the ones chosen by a human and appears to select coherent breakpoints. An interesting observation is that our algorithm tends to merge very short consecutive paragraphs. Moreover, our method of modeling paragraphs as a clustering task can provide insights into how well a written work is structured. In an applied setting, learners might receive suggestions on an alternative paragraph structure that is topologically more cohesive and potentially separable in terms of ideas.

The proposed framework also opens the possibility of further, more complex, use cases. For now, the order of sentences is maintained in the task of paragraph selection. However, the same technique could be applied to determine whether a sentence is well suited in its paragraph, or whether a portion of a text is ill-placed and should be removed or placed within a different paragraph. Thus, follow-up extensions envision the identification of out-of-context sentences and the generation of potential suggestions for sentence reordering.

Ultimately, our objective is to guide learners toward improving their writing through automated feedback on how to better and more clearly express their ideas. The feedback introduces key positions where to introduce paragraph breaks to enhance the overall organization of the text. Beyond providing feedback to learners, our method is also a first and essential step towards improving text structure automatically, by accounting for the maximization of text cohesion, both globally at the paragraph level, and locally, between sentences within paragraphs. Overall, the research reported here represents an initial step in developing an algorithm that can be used to guide students as to how to better organize their writings. However, we have not explored the usage of such a system in an educational setting within the context of this paper, with such an experiment representing an important future step for the work detailed so far.

Acknowledgments

This research was supported by a grant from the Romanian National Authority for Scientific Research and Innovation, CNCS – UEFISCDI, project number TE 70 PN-III-P1-1.1-TE-2019-2209, ATES – “Automated Text Evaluation and Simplification”, the Institute of Education Sciences (R305A180144 and R305A180261), and the Office of Naval Research (N00014-17-1-2300; N00014-20-1-2623). The opinions expressed are those of the authors and do not represent the views of the IES or ONR.

References

1. Stark, H.A.: What do paragraph markings do? Discourse processes, 11(3), 275–303 (1988)
2. Darvishy, A., Nevill, M., Hutter, H.-P.: Automatic paragraph detection for accessible PDF documents. In: Int. Conf, on Computers Helping People with Special Needs, pp. 367–372. Springer, Linz, Austria (2016)
3. Roscoe, R.D., Varner, L.K., Crossley, S.A., McNamara, D.S.: Developing pedagogically-guided algorithms for intelligent writing feedback. International Journal of Learning Technology 25, 8(4), 362–381 (2013)
4. Genzel, D.: A Paragraph Boundary Detection System. In: Int. Conf. on Intelligent Text Processing and Computational Linguistics pp. 816–826. Springer, Berlin, Heidelberg (2005)
5. Sporleder, C., Lapata, M.: Automatic paragraph identification: A study across languages and domains. In: Int. Conf. on Empirical Methods in Natural Language Processing, pp. 72–79. ACL, Barcelona, Spain (2004)
6. Dascalu, M., McNamara, D.S., Trausan-Matu, S., Allen, L.K.: Cohesion Network Analysis of CSCL Participation. Behavior Research Methods, 50(2), 604–619 (2018)
7. Roscoe, R.D., Varner, L.K., Weston, J.L., Crossley, S.A., McNamara, D.S.: The Writing Pal intelligent tutoring system: Usability testing and development. Computers and Composition, 34, 39–59 (2014)
8. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient Estimation of Word Representation in Vector Space. In: Workshop at ICLR, Scottsdale, AZ (2013)
9. Davies, M.: The 385+ million word Corpus of Contemporary American English (1990–2008+): Design, architecture, and linguistic insights. International journal of corpus linguistics, 14(2), 159-190 (2009)
10. Arora, S., Liang, Y., Ma, T.: A simple but tough-to-beat baseline for sentence embeddings. In: 5th Int. Conf, on Learning Representations (ICLR 2017), Toulon, France (2017)
11. Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th Int. Conf. on Machine Learning (ICML-10), pp. 807–814 (2010)
12. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation, 9(8), 1735–1780 (1997)
13. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv, preprint arXiv:1412.6980 (2014)
14. Loshchilov, I., Hutter, F.: SGDR: Stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983 (2016)
15. Rousseeuw, P.J.: Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 20, 53–65 (1987)