BEÜ Fen Bilimleri Dergisi
10 (4), 1349-1365, 2021
DOI: 10.17798/bitlisfen.949052

BEU Journal of Science
10 (4), 1349-1365, 2021

Araştırma Makalesi / Research Article

CatSumm: Extractive Text Summarization based on Spectral Graph Partitioning and Node Centrality

Taner UÇKAN[1*], Cengiz HARK[2], Ali KARCİ[3]

[1] Van Yüzüncü Yıl Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Van*
[2]Turgut Özal Üniversitesi, Mühendislik ve Doğa Bilimleri Fakültesi, Yazılım mühendisliği Bölümü, Malatya
[3]İnönü Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Malatya
(ORCID: 0000-0001-5385-6775) (ORCID: 0000-0002-5190-3504) (ORCID: 0000-0002-8489-8617)

Abstract
In this paper, we introduce CatSumm (Cengiz, Ali, Taner Summarization), a novel method for multi-document document summarisation. The suggested method forms a summarization according to three main steps: Representation of input texts, the main stages of the CatSumm model, and sentence scoring. A Text Processing software, is introduced and used to protect the semantic loyalty between word groups at stage of representation of input texts. Spectral Sentence Clustering (SSC), one of the main stages of the CatSumm model, is the summarization process obtained from the proportional values of the sub graphs obtained after spectral graph segmentation. Obtaining super edges is another of the main stages of the method, with the assumption that sentences with weak values below a threshold value calculated by the standard deviation (SD) cannot be included in the summary. Using the different node centrality methods of the CatSumm approach, it forms the sentence rating phase of the recommended summarising approach, determining the significant nodes and hence significant nodes. Finally, the result of the CatSumm method for the purpose of text summarisation within the in the research was measured ROUGE metrics on the Document Understanding Conference (DUC-2004, DUC-2002) datasets. The presented model produced 44.073%, 53.657%, and 56.513% summary success scores for abstracts of 100, 200 and 400 words, respectively.

Keywords: Document summarization, Summarization, Extractive summarization, Spectral partitioning, Graph-based summarization, Edge Reduction

CatSumm: Spektral Çizge Bölmeleme ve Düğüm Merkeziliklerine Dayalı Çıkarıcı Metin Özetleme

Öz
Bu çalışmada, çok belgeli metin özetleme için yeni bir yöntemi CatSumm (Cengiz, Ali, Taner Özetleme) tanıtılmaktadır. Önerilen yöntem, üç ana adıma göre bir özet oluşturmaktadır: Giriş metinlerinin temsili, CatSumm modelinin ana aşamaları ve cümle puanlama. Girilen metinlerin gösterimi aşamasında kelime grupları arasındaki anlamsal bağlılığı korumak için bir Metin İşleme yazılımı tanıtılmış ve kullanılmıştır. CatSumm modelinin ana aşamalarından biri olan Spektral Cümle Kümeleme (SCK), spektral çizge bölmeleme sonrasında elde edilen alt çizgelerin oransal değerlerinden elde edilen özetleme işlemidir. Standart sapma ile hesaplanan bir eşik değerinin altında kalan cümlelerin özete dahil edilemeyeceği varsayımıyla, yöntemin ana aşamalarından bir diğeri de süper kenarların elde edilmesidir. Son olarak, araştırma kapsamında metin özetleme amacıyla CatSumm yönteminin sonucu, Belge Anlama Konferansı (DUC-2004, DUC-2002) veri setleri üzerinde ROUGE metrikleri ile ölçülmüştür. Sunulan model 100, 200 ve 400 kelimelik özetler için sırasıyla %44.073, %53.657, %56.513 özet başarı puanı üretmektedir.

Anahtar kelimeler: Belge özetleme, Özetleme, Çıkarıcı özetleme, Spektral bölmeleme, Çizge tabanlı özetleme, Kenar azaltma

[*]Sorumlu yazar: taneruckan@yyu.edu.tr
Received: 07.06.2021, Accepted: 02.07.2021

1. Introduction

Raw data are information communities that are not yet fully revealed in relation to each other, while they can also be defined as movable strings which can be expressed in digital formats. This data needs to be analyzed with the aim of converted into meaningful and useful data resources. It is hence necessary to develop novelty methods with the aim of reduce the time required for access this data to an acceptable level, as well as to analyze the data [1- 6].

Automated text summarization is one such method of analysis. In many fields such as business, academia, and healthcare, the ability to summarize is essential, and therefore text summarization is still an important area of study for academic researchers. In 2002, Radev [7] referred to a summary as texts which were often shorter than the original text or texts, but not significantly more than half the inventive document or texts. In 2004, Erkan [8] defined text summation as the method of producing a form of specific document that can still ensure beneficial data to the consumer. In 2007, Das [9] defined automatic text summarization as a significant and short form of a text with the help of machines without need for human intervention. In 2019, Joshi [7] identified document summarisation as an important element aimed at representing texts in a compact form. The knowledge content that a summary should be able to carry can be specified by the user. Subject-oriented summaries can focus on the user's orientation, and this way documents can be summarized. General summaries preserve the general content of the main document and aim for maximum information coverage. [8, 10].

In general, there are two different type in document summarisation systems. Extractive summarisation systems select considerable blocks from the main text. Extractive summarisation systems weight sentences with a set of predefined properties. The rated sentences or clauses are sorted and summarization of the text units with the highest score and the required size is formed. The aforementioned system consists of leaner steps than the abstractive summarizing system. Abstractive summarizing systems redefine the sentences by way of interpretation. This approach is to interpret the main text and restate it with fewer words. When using the linguistic methods for understanding and interpreting texts, the method embraces new concepts and expressions to extract important information from the main document; and may consist of either form of text summarization. For all these reasons, extractive summation systems are more flexible and applicable [2, 11– 14]. In addition, document summarization approaches can be classified as single or multi text [7, 12- 13].

Recently, powerful graph-based approaches to document summarisation have been suggested in NLP. Mihalcea [18], his approach called TextRank, summarized texts with the help of graph-based sentence scoring. Likewise, with the LexRank approach, Erkan [8] calculated points scored in performing extractive summaries. Moreover, Parveen [19] introduced the Egraph. In 2020, Hark and Karci [20] presented a new model based on graph and entropy. In addition, Uçkan and Karci [21] presented another innovative study using independent sets in graphs. In the current study, we aim to take unsupervised graph-based aproach in extractive document summarisation one step further.

The next parts of the work are organized as follows: Section 2 considers similar studies to be found in the literature that are related to the subject. In Section 3, detailed information of the proposed CatSumm summation method’s stages are given. Section 4 provides information about the data set used and the evaluation criteria used, as well as the empricial results of the suggested text summarisation aproach. Lastly, discusses and interprets the experimental results of the current study.

2. Material and Methods

Document summarization studies date back to the last century. One of the most important and influencing studies in document summarisation was performed by Luhn in 1958. In his study, he recommended weighting the words of the text as a function by ignoring words with the highest frequency [22]. In later studies, summarization models combined new properties like considering the frequency of words the positions of the sentences and hint words in the texts [23]. Document summarisation is divided into two as abstracting and extractive [13]. In extractive summarization, the words or sentences in the text’s abstract are retained without being changed, whereas abstractive summarization explains the basic information expressed in the text by creating different words and sentences [24- 25]. Summarizations made by human hand are mostly not considered to be extractive.

However, most of the published studies have shown that extractive summarization provides better results over abstractive methods. Therefore, research on the subject of summarization has generally concentrated on extractive summarization [8]. In current studies, a preprocessing tool is used. [26]. After pretreatment, various techniques have been employed in summarization studies, with the main purpose of these studies being to find the most valuable sentences that should be included in the abstract of a given text.

Various scoring methods are applied when selecting sentences in texts. In graph-based scoring methods, the scores assigned to sentences are based on their interrelationships. When one sentence refers to another, it is considered that a strong relationship exists between them. Graph-based scoring algorithms such as HITS [30] or PageRank [31] are used in many different fields. Another method developed for graph-based scoring is the TextRank method [18], in which the importance of words in the text are determined as graph-based. Similarly, by using graph-based methods, sentences in texts and their common word weights are considered in determining the importance levels of sentences [27]. Diagrams created with the LexRank method are expressed as matrices, with the Eigenvector Centrality values of these matrices used to determine the importance of sentences [8]. In similar studies, spectral graph partitioning techniques are used in order to use the graphs obtained from the texts more effectively and to choose the correct sentences [28, 29].

2.1. Proposed Method (CatSumm)

Graphical illustration of CatSumm approach is shown in Fig.1. The presented The CatSumm document summary approach has three layers. Firstly, some preprocesses are applied by using a text preprocessing tool that was developed named KUSH. Next layer, the relations among the sentences are symbolized in a formal and the nodes that are formed by the Spectral Sentence Clustering (SSC) method are divided into groups. Edge reduction is then applied with standard deviation so that lines with super edges are obtained where strong relationships are more pronounced. In the third and final stage, different nodes centrality approaches of the super-edge graphs are applied, with important nodes and thus important sentences identified.

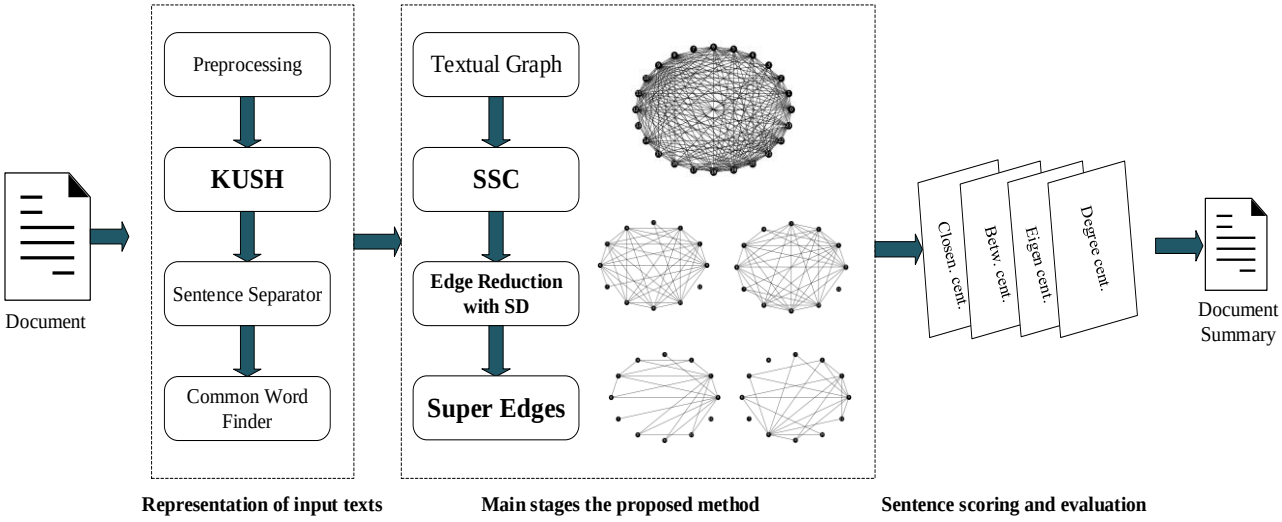

Figure 1. Schematic outline of proposed CatSumm model for text summarization

2.2. Kush Text Processing Tool

The KUSH text processing tool forms one of the representative stages for texts to be summarized, and is therefore considered the text processing and preparation layer of the presented document summarisation method. Prior to this stage, it is difficult to determine the connection and interrelations between sentences where words that differ with regard to spelling are derived from a joint vocable origin. Intuitively, this was predicted to significantly affect performance achieved after classification, and the document preprocessing software was developed for application before the Sentence Separator step within the scope of the presented CatSumm. The KUSH was developed on .NET platform using C#. The pseudocode of the approach is presented in Algorithm 1.

Table 1 contains the transformations that the KUSH software tool performed on the DUC-2002-d070f text document. Documents to be summarized are simplified based on the common words they contain. There is no simplification applied for words that have no intersection in the text. With this dynamic working principle, the KUSH software tool redefines text parts to be simplified each time depending on the texts to be classified. In this way, as the texts change, it simplifies different words. In Table 1, the words that are modified or deleted are shown in bold font.

Algorithm1. KUSH algorithm

Step 1	***(Input)*** Inputs obtained from the DUC datasets are presented as input data to the algorithm.
Step 2	***(Preparation)*** The matrix and variables to be used are defined and initial values assigned.
Step 3	***(Sentence vectored)*** The text is separated.
Step 4	**(*Word vectored)*** Word vectors are created from sentence vectors.
Step 5	***(All alternatives)*** The most suitable word is selected from the alternative list (based on n-gram)
Step 6	***(Best alternatives)*** The most suitable alternative is selected from the alternative words.
Step 7	(***Change word vector)*** Replace the most appropriate alternative found in the word vectors.
Step 8	***(Loop)*** Repeat until the word vector size.
Step 9	***(Create output text***) Combine words in the word vector according to the order in the sentence vector after all operations have ended.

Table 1. Two text conversions from the d070f

		Text before KUSH preprocessing	Text after KUSH preprocessing
Word-based	*1*	*Lawyer, powerful, husband's*	*Law, power, husband*
	2	*Russian- Chilean- newspaper*	*Russia- Chile- news*
	Sample text from DUC-2002		
Sentence-based	*1*	*Erich Honecker,* ***the*** *former GDR head* ***of*** *state, died* ***at his*** *house* ***in*** *Santiago, Chile* ***on*** *Sunday morning [29 May], according* ***to his lawyer****.*	*erich honecker former gdr head died house santiago chile sunday morning [29 may] according* ***law****.*
	2	***Lawyer*** *Nicolas Becker,* ***who had*** *represented 81-year-old Honecker* ***before the*** *Berlin court in 1992* ***and early*** *1993, told DPA* ***on*** *Sunday afternoon* ***that*** *Honecker* ***had*** *rejected* ***an*** *operation.*	***Law*** *nicolas becker represented 81 year old honecker berlin court 1992 1993 told dpa sunday afternoon honecker rejected operation.*
	3	***However****,* ***everything that was*** *sensible* ***had*** *been* ***done with*** *regard* ***to his*** *cancer.*	*sensible regard cancer.*

As explained in this study, texts with the characteristics of raw and everyday spoken language are primarily eliminated as unnecessary and undesired data with the text preprocessing and preparation process. Then, the developed KUSH tool prevents expressions with similar or very similar meanings which are perceived as having different meanings when creating the graphs. Considering the SSC model is semantically distinguishable between the sentences and provides healthier measurable relationships, this shows that tests conducted with the KUSH tool is effective in catching these relationships.

2.3.Textual Graph

Modeling of the problems can be realized through graphs by the formal representation of the features and the relations between these features. The graphs are conceptually composed of the nodes and the edges representing the relations between the nodes. Nodes and edges are two finite sets. Nodes are the main individuals that form the group represented by the graph. The edges are the relevant relationships between the main individuals. Generally, the graph is shown as $G=(V, E)$. The set of nodes is $V=\{v_1, v_2,...,v_n\}$, and the set of edges is $E=\{e_1,e_2,...e_n\}$ $(E\subseteq VxV)$. The terminal nodes for the edge $e_i=\{v_j,v_{j+1}\}$ are v_j and v_{j+1} node. For the neighbor nodes v_j *and* v_{j+1}, if $e_1=(v_j, vj_{+1}) \in E$ and $(v_{j+1},v_j)\in E$, these are non-oriented edges. The graphs for these kinds of edges are non-oriented graphs. If v_j *and* v_{j+1} neighbor nodes are $e_1=(v_j, vj_{+1}) \in E$ and $(v_{j+1},v_j)\notin E$, these kind of edges are oriented edges. A graph that is formed by oriented edges is called an oriented graph [2,30]. If the edges $E=\{e1,e2,...en\}$ that represent the relationship between the nodes V on a $G=(V, E)$ graph $E=\{e_1,e_2,...e_n\}$ carry nonnegative weights, this kind of graph is called a weighted graph [31]. In the current study, the graphs created to represent the texts are weighted.

There are different approaches in the literature in order to represent textual graphs. While a full sentence or a clause different relationship forms, such as the intersection between the nodes and the co-occurrences, can be represented by the edges. D can be also represented as a set containing the m sentences from the documents in the collection, i.e., $D = \{s_1, s_1, \dots, s_m\}$. In this case, Equation 1 and Equation 2 are taken into account in order to find the similarity between the sentences.

$$Sim(s_i, s_j) = \sum s_i \cap s_j \tag{1}$$

$$E = \begin{cases} Count(w) & if\ s_i \cap s_j \neq \emptyset \\ 0 & if\ s_i \cap s_j = \emptyset \end{cases} \tag{2}$$

These different forms of expression have both advantages and disadvantages. As Karcı stated [30], the ability to dynamically express the graphs provides the advantage of memory usage and the ability to interfere with the number of nodes and edges during the study. However, it also brings some difficulties in terms of its use in response to these advantages. The memory is reserved for each node and edge.

In the current study, the preference was to represent the graphs with matrices. After the representation of the texts to be summarized, the proposed CatSumm model creates graphs representing the texts in order to apply spectral graph partitioning methods on data with a textual format. The operations performed in this section correspond to the Textual Graph stage from the steps schematized in Figure 1. Figure 2 contains a simple text sample and a weighted and non-directional Sentence-Word Graph created for this text. A node is added for each sentence in the text. Weights are added to the edges between nodes (see Equation 1). Where there were no intersecting words between the sentences, no edges were added between the relevant nodes (see Equation 2).

2.4. Spectral Sentence Clustering (SSC)

Suggested in the study SSC method separates the nodes of the generated line into groups. This approach associates each sentence that makes up the mainline with other sentences. Using this relationship, sentences are clustered. With the KUSH software tool, these clusters are aimed to reach a more accurate and distinct structure. Although used in different fields, we use spectral graph theory techniques to analyze the general structural features of the graph on the basis of graphs. This technique aims to divide the graph nodes (at least or at most) as possible to the equivalent groups which do not intersect at least the cutting cost by considering the distance or weights between the nodes. In the proposed method, we use the techniques of spectral graph theory for clustering the sentences of the texts according to their subjects. In this study, the Laplacian matrix is used when applying spectral graph partitioning method. The Laplacian matrix of a graph carries a lot of information about the graph, just like the neighborhood

matrix, but has many different uses and different specifics. In this study, Simple Laplacian matrix was used. For a given G graph, the matrix D represents the matrix of degrees, the matrix A represents the adjacency matrix. In this case, the Simple Laplacian matrix is calculated as in Equation 3

$$L = D - A \ , \ d_i = \sum_{\{j|(i,j)\in E\}} w_{ij} \tag{3}$$

The values of the L matrix are given as in Equation 4.

$$LaplaceG(i,j) = \begin{cases} \sum_{(i,k)\in E} A(i,k) & , \quad if \ i = j \\ -A(i,j) & , \quad if \ i \neq j \\ 0 & , \quad other \end{cases} \tag{4}$$

The edges bind the nodes together and if there is a connection between these two nodes, it is considered $(u, v) \in E$. The neighborhood matrix of the G graph is $A(G)$. $A(G)$ matrix is defined as in Equation 4. By using the edges between the nodes, the Adjacency matrix ($A\ (G)$) of the corresponding graph is obtained (see Equation 5). The degree matrix ($D\ (G)$) of the graph G (see Equation 6) is obtained from the number of all the edge numbers found in a node.

$$A_{i,j} = \begin{cases} w_{i,j} & , \quad (i,j) \in E \\ 0 & , \quad (i,j) \notin E \end{cases} \tag{5}$$

$$D_{i,j} = \begin{cases} d_{i,j} & , \quad i = j \\ 0 & , \quad i \neq j \end{cases} \tag{6}$$

The coarse code of our clustering method is as shown in Algorithm 2. The basic idea in the algorithm is to group the nodes of the text documents that correspond to the sentences on the basis of their interrelationships. Thus, it is based on the foresight that the sentences in the same category may be similar, and that sentences in different categories may be less similar or are not similar at all. The way that the SSC approach works is schematized on an example graph. Figure 3 shows the nodes of representative clusters obtained. The sentences in the same category as the SSC model are more closely related to each other in terms of the sentences in different categories.

Algorithm 2. SSC: spectral sentence clustering algorithm

Step 1	***(Input)*** Texts obtained from the KUSH software tool are presented as input data to the algorithm.
Step 2	***(Preparation)*** The matrix and variables to be used are defined and initial values assigned.
Step 3	***(Sentence vectored)*** Text is separated..
Step 4	***(Word vectored)*** Word vectors are created from sentence vectors.
Step 5	***(intersection (sentence(i) ∩ sentence(j))*** The Adjacency matrix and the degree matrix are obtained by calculating the number of common words between sentences.
Step 6	***(Laplacian transformation)*** The Laplacian matrix is calculated using the Simple Laplacian method (see Equation 3
Step 7	***(Eigen decomposition)*** The resulting Laplacian matrix is used and Eigen decomposition is used to obtain eigenvalues and eigenvectors.
Step 8	***(Fiedler's Theory)*** Eigenvalues are ordered from large to small, then the vector of the second smallest eigenvalue is obtained [32].
Step 9	***(Clustering)*** The obtained eigenvector value is checked and the values with the same sign (+ or -) are collected in a cluster.

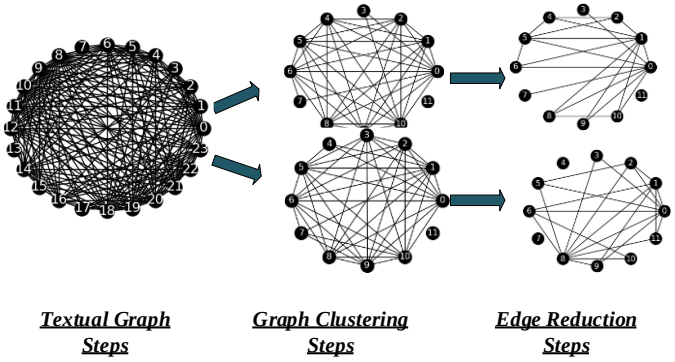

Figure 3. Example document in DUC-2002 dataset shows edge connections before and after standard deviation of sub-graph belonging to a sample sentence

2.5. Graphical Simplification with Standard Deviation

This step describes the elimination of weak connections from the two sub-graphs obtained by Spectral Sentence Clustering from the main graph by standard deviation. The nodes in the sub-graphs each represent one sentence. The higher the relationship between two sentences, the higher weight of the edge. In this study, it was found that sentences associated with the valuable sentences were also valuable, and therefore included in the abstract. For this reason, the standard deviation values of all relations between the sentences were calculated and those relations that fell below this value were reduced and eliminated.

Standard deviation is a statistical method that describes how close various data are to the mean in a data set [33]. With standard deviation, we find out how much of the data is close to the average. As V represents the nodes and E the sides, on a graph which is $G = (V, E)$ the weight of sides is $W = \{w_1, w_2, w_3, \ldots, w_n\}$ and the arithmetic average is shown as $\overline{w}$.

$$\overline{w} = \frac{\sum_{i=1}^{N} w_i}{N} \tag{7}$$

$$P(e_1) = \frac{w_1}{\sum_{i=1}^{n} w_i} \tag{8}$$

$$Var(W) = \sum_{i=1}^{n} (w_i - \overline{w})^2 * P(e_i) \tag{9}$$

$$\sigma = \sqrt{Var(W)} \tag{10}$$

Standard deviation value σ is calculated by using Equations 7-10. With the calculated standard deviation value, the edge weights which can be neglected are determined. The edge weights are reduced by the edge values with a distance above the standard deviation of the arithmetic mean. Therefore, the most valuable links we call the Super Edge are obtained and the sentences to be found in the abstract are made up of stronger sentences. As shown in Figure 3, it is observed that the strong relations are more distinct when edge reduction is applied to the graphs obtained by standard deviation. After applying the edge reduction to the graphs, the sentences which are found in the abstract are selected according to the obtained measurement values by applying node centrality measurements to the graph. The Node Centrality methods are described in detail in the next step.

2.6. Node Centrality Measures

The concept of centralization has been proposed by [34] to determine the position of an individual and its impact on group-wide processes. Since then, many centrality measurements have been proposed in the literature. Each of these measures advocates different ideas about what it means to be "centralized" within a network. Centrality measurements were used to identify central points or central nodes in many different areas [8, 11, 35]. In the graphs obtained from texts in the current study, the centralized measurements of Betweenness Centrality (see Equation 11), Closeness Centrality (see Equation 12), Degree Centrality (see Equation 13), and Eigenvector Centrality (see Equation 14) have been used to determine those most valuable among the nodes representing the sentences. Characteristic information about the node centrality measures is presented in Table 2.

Table 2. Description of node centrality measure

Node Centrality Measure	Formulation	
Betweenness Centrality [2]	$C_b(v) = \sum_{x,y \in N} \frac{G_{x,y}(v)}{G_{x,y}}$	(11)
Closeness Centrality [2, 36, 37]	$C_c(v_i) = \frac{N-1}{\sum_{v_j \in v} distance(v_i v_j)}$	(12)
Degree Centrality [38–40]	$C_D(v) = \frac{deg(v_i)}{N-1}$	(13)
Eigenvector Centrality [36]-[37], [44]	$C_e(V_i) = \frac{1}{\lambda} \sum_{V_j \in N(V_i)} V_{ji}\, x\, C_e(V_j)$	(14)

3. Results and Discussion

In this part of the study, the dataset used during the experimental processes to test the summarized method is described. Popular types of evaluation criteria are also used to evaluate the accuracy of summarization systems. Finally, a series of test results are introduced in evaluating the efficiency of the suggested summarisation approach.

3.1 Dataset

In the approach presented, the DUC-2002 and DUC-2004 datasets were tested. The datasets contain documents for summarisation. In this study, summarisation were used as the presented model utilizes an extractive summarization method [41]. Characteristic information about these datasets is given in Table 3.

Table 3. Characteristics of the datasets

Description	DUC-2002	DUC-2004
Number of clusters	59	50
Number of documents in each cluster	~10	10
Number of documents	567	500
Summary length	200 and 400 words	665 bytes

3.2 Evaluation Metric

The evaluation criterion used is ROUGE performance [42, 43]. In this study, we use ROUGE(N-L-W1.2-SU) measures to evaluate the performance of the proposed CatSumm approach. ROUGE-N evaluates the number of n-grams shared between the created and golden summary.

$$ROUGE - N = \frac{\sum_{C \in \{Ref\}} gram_n \in S \sum Count_{match}(gram_n)}{\sum_{C \in \{Ref\}} gram_n \in S \sum Count(gram_n)} \quad (15)$$

where N in the formula is equal to the length of $gram_n$ n-gram, and $Count_{match}(gram_n)$ are the maximum number of n-grams intersecting in the abstract and reference summary. Equation 15 clearly shows that ROUGE-N is a related measurement with Recall as the denominator of equality is the sum of the number of n-grams generated in the golden summary [42]. For instance (N-1) measures the number of uni-grams shared between two summaries. In the same way (N-2) calculates the number of bigrams that intersect between the suggested and the golden summary. Similarly, the ROUGE-L value focuses on the longest common sequence. As X and Y are given by the two series of words, calculations for ROUGE-L values were made in Equations 16-18 for the series.

$$R_{lcs} = \frac{LCS(X,Y)}{m} \quad (16)$$

$$P_{lcs} = \frac{LCS(X,Y)}{n} \quad (17)$$

$$F_{lcs} = \frac{(1+\beta^2)R_{lcs}P_{lcs}}{R_{lcs} + \beta^2 P_{lcs}} \quad (18)$$

ROUGE-W-1.2 computes the matches that occur consecutively between the suggested and the golden summary. ROUGE-SU calculates the bigrams.

3.3. Experimental Studies

The primary purpose of the current research is to present an uncontrolled and graph-based process in order to summarize the extractive text by moving text summary studies one step forwards.

To evaluate the achievement of the CatSumm summarisation system, empiricial studies were performed by presenting summaries of texts from the DUC datasets. In this study, to improve the achievement of the CatSumm method, a preprocessing tool was developed and the texts were subjected to certain preliminary procedures with specific linguistic processes. The success of the summaries were then measured by calculating the most commonly used performance metrics found in the literature. The efficiency of the introduced was then compared to other summarisation approachs.

In the experimental study, the steps mentioned in Figure 1 were followed. Primarily, stop words, that is unwanted and non-representative expressions and characters, were excluded from the dataset. As shown in Figure 1, a document preprocessing was introduced and used immediately after the preprocessing step and before the Sentence Separator stage. The KUSH text processing software contributes to the success and robustness of the CatSumm text summarization method, as in the experimental studies without using the KUSH software, high success values could not be obtained.

Although different Laplacian calculation methods are available in the literature, the Simple Laplacian calculation was preferred in this study. The eigenvalue and eigenvalue vector pairs obtained using the generated Laplacian matrices are listed. With this listing, the Laplacian spectrum, which belongs to the textual graph and carries information about the connectivity of the graph, has been formed. The eigenvalue vectors corresponding to the second smallest eigenvalue represent the algebraic commitment of the graph. In the experimental study, this vector was used to divide the graph. In this context, in order to increase the summarizing performances of the texts represented by the graphs,

summaries were obtained by using the SSC by dividing the graphs representing the abstract and by the ratios obtained after spectral portioning. Quite successful results are reported with this new approach.

In addition to these steps, a threshold value was calculated with a certain standard deviation calculation in order to obtain the most important nodes in the segmented graphs. By separating the edges below the calculated threshold value from the graph, the sentences that represent the relevant edges and nodes were excluded from the summary. Different Node centrality calculations of the nodes connected with the super edges obtained at the last stage were then calculated (Degree, Closeness, Betweenness, Eigenvector) and summaries of 100, 200, and 400 words for each approach were obtained.

To evaluate the summarizing efficiency of the study, we conducted a sample summary (100, 200, and 400 words) of the texts in the DUC (2002- 2004) datasets, and system summaries obtained by the CatSumm method were thereby compared. For this purpose, ROUGE summarizing performance metrics were employed to measure the success of the summarisation results. In the empricial study, Recall values of ROUGE-1, 2, L, W-1.2, SU metrics were calculated for evaluating the summarization success of the introduced system summary.

The CatSumm model was first run separately for selected node centrality values and different summaries were obtained. Recall values of the obtained 200-, 400-, and 100-word abstracts are shown in Tables 4, 5 and 6. It can be seen from the tables, the eigenvector centrality value yielded the best results for the 200- and 100-word abstracts, and the second-best results for the 400-word abstract. Considering these results, it is understood that the most appropriate centrality value for the proposed model is the eigenvector centrality value. In Table 7-9, the CatSumm + Eigenvector Centrality values that outperformed other approachs have been highlighted in bold font. As can be observation from the tables, the proposed method outperformed all other methods in summaries of 100 and 200 words, and very competitive results were observed for the 400-word summary.

Table 4. Recall values of 200-word summarization on DUC-2002 dataset using node centrality values in proposed CatSumm summarization method

ROUGE evaluation methods	CatSumm+ Betweenness Centrality	CatSumm+ Closeness Centrality	CatSumm+ Eigenvector Centrality	CatSumm+ Degree Centrality	Average
ROUGE-1	0.49421	0.52407	**0.53657(1)**	0.52628	0.52028
ROUGE-2	0.20106	0.22930	**0.26097(2)**	0.24883	0.23504
ROUGE-L	0.44847	0.48566	**0.50195(1)**	0.48968	0.48144
ROUGE-W-1.2	0.15715	0.17091	**0.17990(1)**	0.17388	0.17046
ROUGE-SU	0.21778	0.23640	**0.24432(1)**	0.23347	0.23299

Table 5. Recall values of 400-word summarization on DUC-2002 dataset using node centrality values in proposed CatSumm summarization method

ROUGE evaluation methods	CatSumm+ Betweenness Centrality	CatSumm+ Closeness Centrality	CatSumm+ Eigenvector Centrality	CatSumm+ Degree Centrality	Average
ROUGE-1	0.56301	0.56414	0.56513 (2)	**0.57381(1)**	0.56652
ROUGE-2	**0.28698**	0.26789	**0.28009(1)**	0.27707(2)	0.27800
ROUGE-L	0.53151	0.53119	0.53749(2)	**0.54205(1)**	0.53556
ROUGE-W-1.2	0.16624	0.16702	0.16805(2)	**0.17033(1)**	0.16791
ROUGE-SU	0.28337	0.28396	0.27937(2)	**0.29344(1)**	0.28503

Table 6. Recall values of 100-word summarization on DUC-2004 dataset using node centrality values in proposed CatSumm summarization method

ROUGE evaluation methods	CatSumm+ Betweenness Centrality	CatSumm+ Closeness Centrality	CatSumm+ Eigenvector Centrality	CatSumm+ Degree Centrality	Average
ROUGE-1	0.37167	0.43553	**0.44073 (1)**	0.43657	0.42112
ROUGE-2	0.05495	0.07876	0.07177 (2)	0.07894	0.07110
ROUGE-L	0.27632	0.32735	**0.34006 (1)**	0.33090	0.31865
ROUGE-W-1.2	0.09405	0.11086	**0.11279 (1)**	0.11110	0.1072
ROUGE-SU	0.12652	0.16541	**0.16939 (1)**	0.16783	0.15728

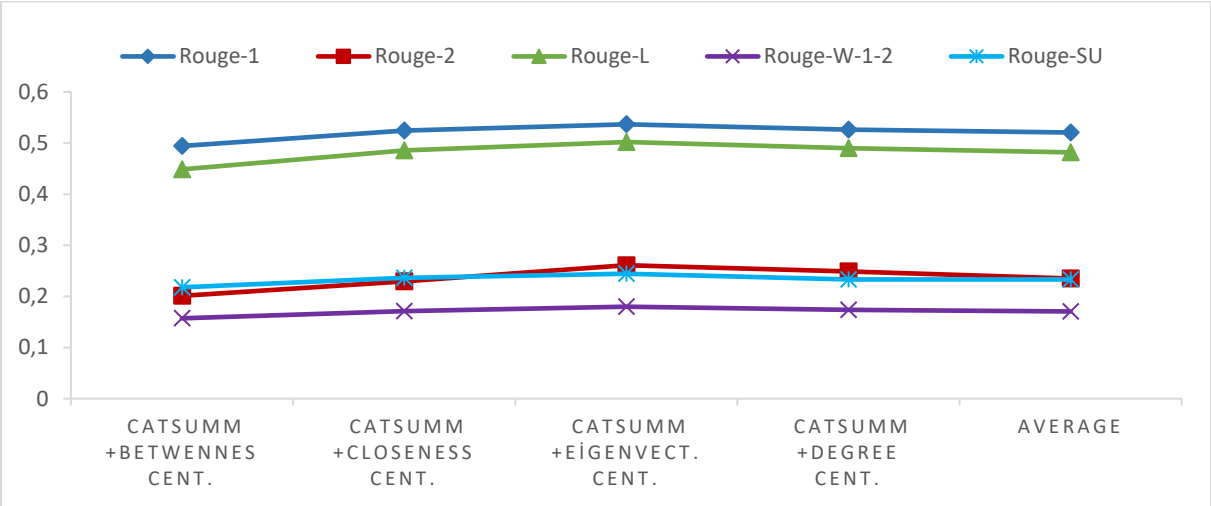

Figure 4. Recall values of 200-word summarization on DUC-2002 dataset using node centrality values in proposed CatSumm summarization method

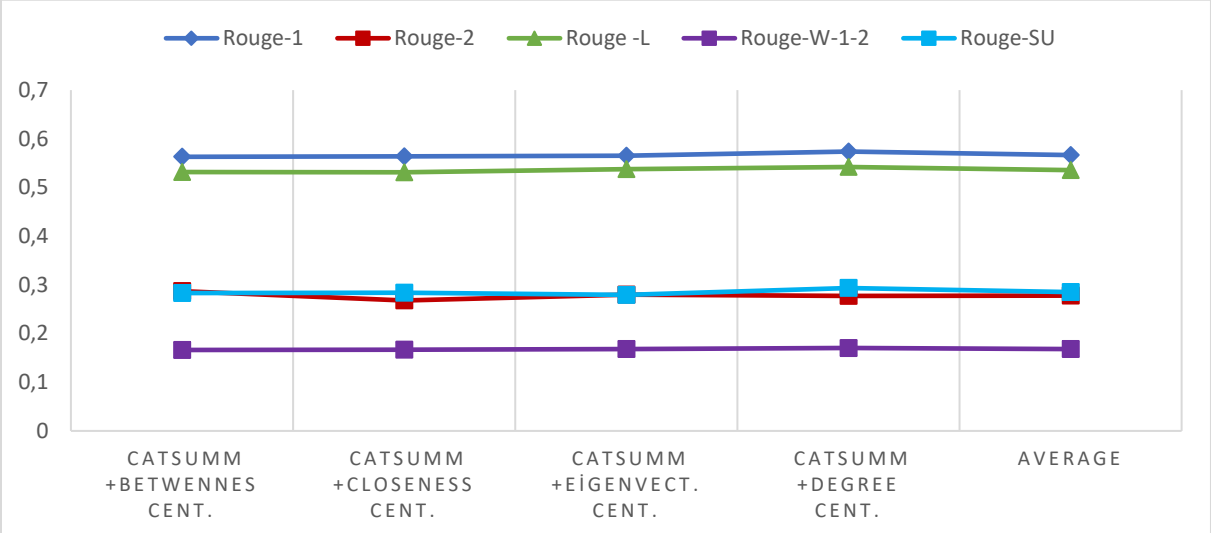

Figure 5. Recall values of 400-word summarization on DUC-2002 dataset using node centrality values in proposed CatSumm summarization method

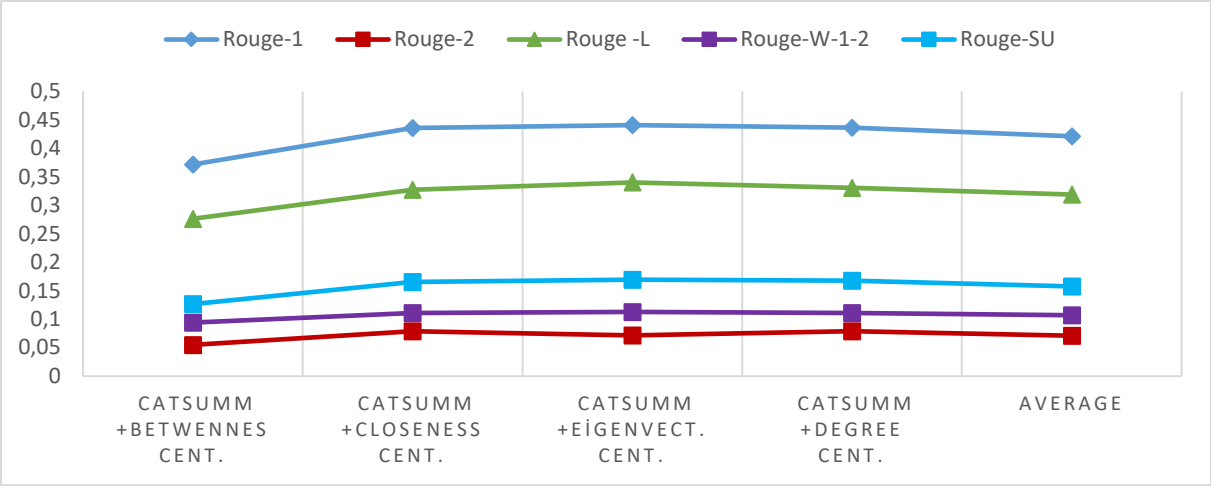

Figure 6. Recall values of 100-word summarization on DUC-2004 dataset using node centrality values in proposed CatSumm summarization method

The Recall values of the ROUGE-(1, 2, L, W-1.2, SU) measurements of the node centrality values used with the CatSumm model for 200-word summaries are demonstrate in Fig 4. As can be observation clearly in Figure 4, when the Eigenvector Centrality measurements are taken into consideration, it is observed that it gives better results than other centrality measurements. Similarly, it is shown in Figure 5 that the Degree Centrality value gives better results with a slight difference and also gives very competitive results in the Eigenvalue value when summarized with 400 words. When the 100-word abstracts made using the DUC-2004 dataset are taken into consideration, it can be seen in Figure 6 that the Eigenvector Centrality values produced the better results. When the results obtained in these studies are taken into consideration in Table 4-6, it is reported that the CatSumm approach produced high values with selected node centrality methods, but the best result was obtained with the Eigenvector Centrality measurement value.

Table 7 and 8 represent the results from the 200 and 400 words summaries of the proposed CatSumm + Eigenvector Centrality summarization model using the DUC-2002 dataset. As can be clearly seen from the tables, our model produced better results than all the other 200-word summaries. For the 400-word abstracts, the CatSumm + Eigenvector Centrality model was observed to produce the second-best result, and shows how competitive the proposed model is to outperforming all the other models. In fact, when looking at the CatSumm + Degree Centrality model in Table 5, it is clear that in most ROUGE measurement values, all the other methods produced lesser results for the 400-word summaries. These results are shown graphically in Figure 7-8. Figure 7 demonstrates the Recall values of the ROUGE(1-2-L-W1.2-SU), for the 200-word summaries obtained using the DUC-2002 dataset. Also, Figure 8 shows the Recall values of the ROUGE(1-2-L-W1.2-SU) for the 400-word summaries obtained using the DUC-2002 dataset.

Table 7. Recall values of 200-word summary of proposed CatSumm + Eigenvector centrality and other summarization methods using DUC-2002 dataset

ROUGE evaluation methods	**Random** [44]	**LSA** [45]	**TextRank** [48], [18]	**LexRank** [8]	**SumBasic** [48]	**KLSum** [49]	**CatSumm + Eigenvec Centrality**
ROUGE-1	0.40932	0.37723	0.48417	0.48124	0.46128	0.37464	**0.53657(1)**
ROUGE-2	0.10562	0.08950	0.16921	0.18507	0.15947	0.12113	**0.26097(1)**
ROUGE-L	0.36328	0.33477	0.46748	0.43813	0.39661	0.34066	**0.50195(1)**
ROUGE-W-1.2	0.12324	0.11295	0.15655	0.15201	0.13782	0.11839	**0.17990(1)**
ROUGE-SU*	0.15112	0.12783	0.21541	0.20321	0.17981	0.13361	**0.24432(1)**

Table 8. Recall values of 400-word summary of proposed CatSumm + Eigenvector centrality and other summarization methods using DUC-2002 dataset

ROUGE evaluation methods	Random [44]	LSA [45]	TextRank [48], [18]	LexRank [8]	SumBasic [48]	KLSum [49]	CatSumm + Eigenvec Centrality
ROUGE-1	0.52799	0.46361	0.57812	0.55018	0.52373	0.43746	0.56513(2)
ROUGE-2	0.20698	0.16001	0.25654	0.23706	0.18916	0.16256	**0.28009(1)**
ROUGE-L	0.48516	0.42283	0.56505	0.51566	0.46814	0.40158	0.53749(2)
ROUGE-W-1.2	0.14844	0.12558	0.17498	0.16162	0.14401	0.12675	0.16805(2)
ROUGE-SU*	0.25020	0.19823	0.31213	0.27351	0.24097	0.17922	0.27937(2)

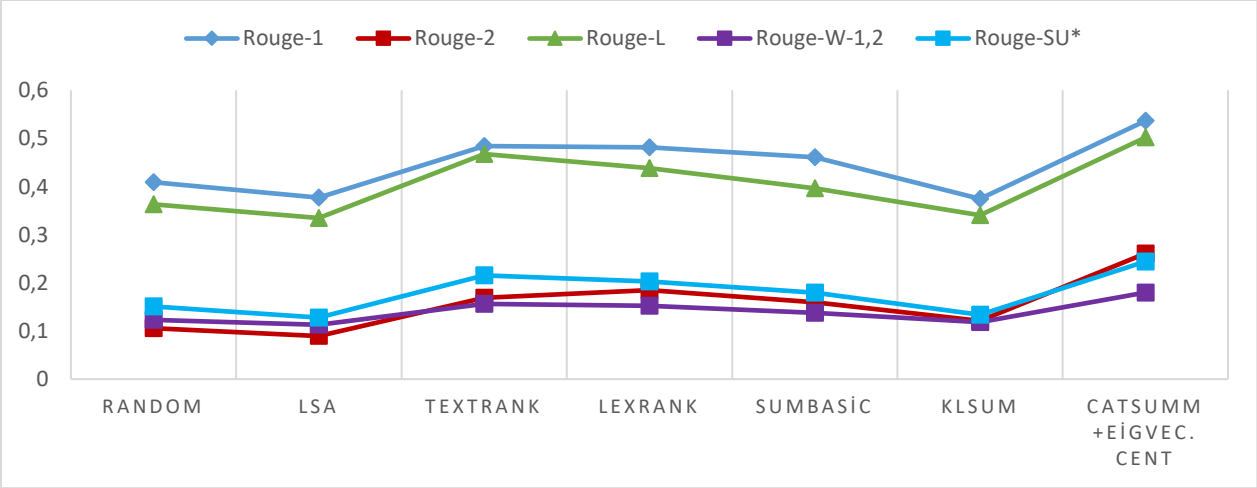

Figure 7. Recall values of 200-word summary of proposed CatSumm + Eigenvector centrality and other summarization methods using DUC-2002 dataset

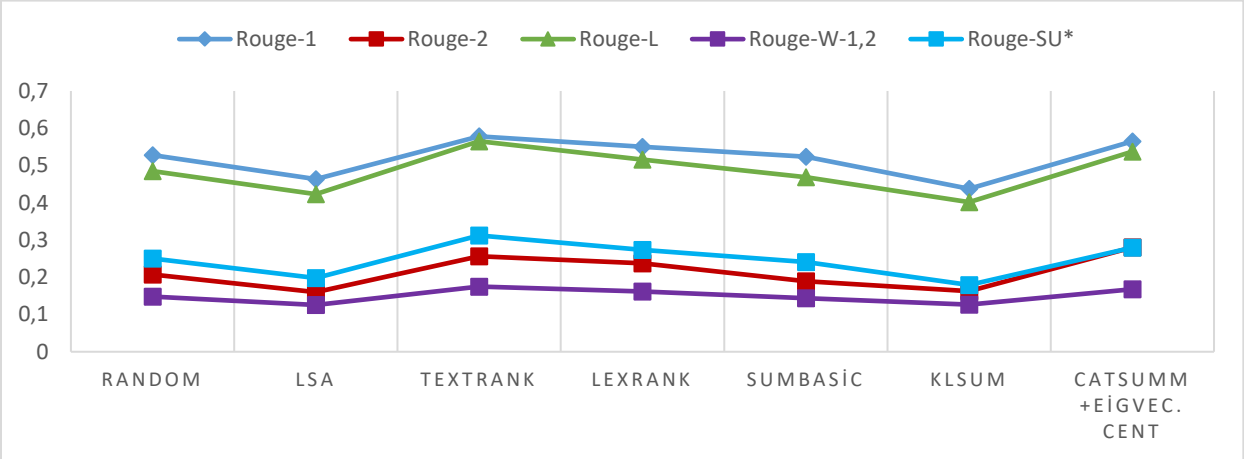

Figure 8. Recall values of 400-word summary of proposed CatSumm + Eigenvector centrality and other summarization methods using DUC-2002 dataset

Table 9 clearly shows the Recall values of the ROUGE(1-2-L-W1.2-SU) for the 100-word summaries obtained using the DUC-2004 dataset. When the reported values are examined, it can be observation the Recall values obtained as a result of the combination of the introduced CatSumm method and the Eigenvector Centrality value produced better results than other summarization methods, except for LexRank's ROUGE-2 value. When the ROUGE-1 Recall values of the proposed model are taken into consideration, it can be seen that it produced results approximately 37% better than Random

value, 57% better than LSA, 10% better than LexRank, and 29% better than SumBasic and KLSum methods.

Figure 9 visually illustrates the Recall values of the (ROUGE-1,2,L,W-1.2,SU) measurement metrics obtained from the different text summarization techniques and the proposed CatSumm summarization method. As can be observed in Figure 9, the CatSumm + Eigenvector Centrality method offers higher Recall-based ROUGE values compared to other summarization methods.

Table 9. Recall values of 100-word summary of proposed CatSumm + Eigenvector centralization and other summarization methods using DUC-2004 dataset

ROUGE evaluation methods	**Random** [44]	**LSA** [45]	**TextRank** [48], [18]	**LexRank** [8]	**SumBasic** [48]	**KLSum** [49]	**CatSumm + Eigenvec Centrality**
ROUGE-1	0.32206	0.27995	0.31301	0.39893	0.33859	0.33778	**0.44073 (1)**
ROUGE-2	0.05439	0.03324	0.04145	0.**07977**	0.05623	0.07030	0.07177 (2)
ROUGE-L	0.25785	0.21295	0.26099	0.30685	0.24408	0.27448	**0.34006 (1)**
ROUGE-W-1.2	0.08957	0.07427	0.08836	0.10436	0.08427	0.09315	**0.11279 (1)**
ROUGE-SU*	0.09532	0.07285	0.09741	0.13998	0.10063	0.10609	**0.16939 (1)**

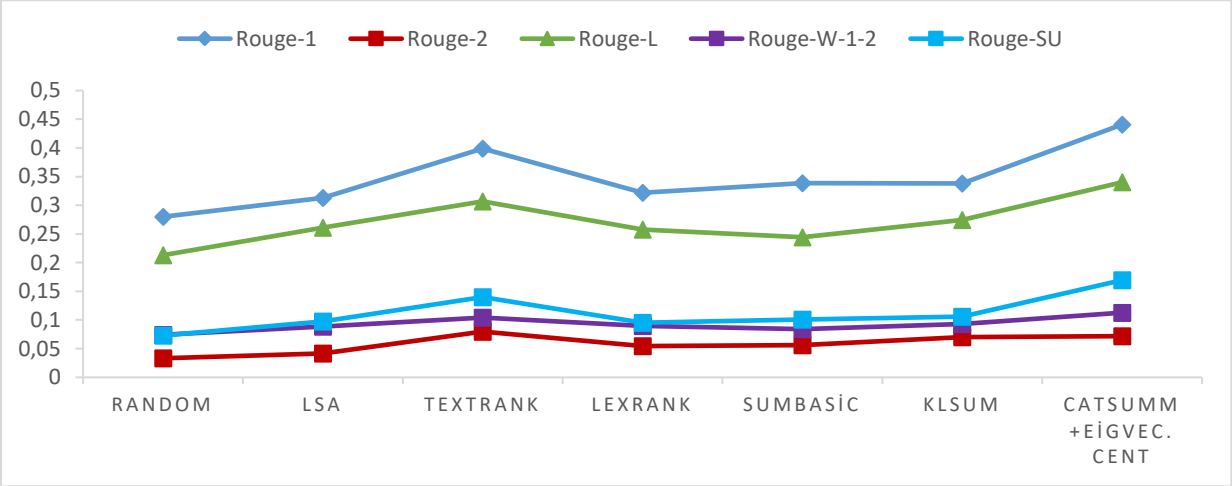

Figure 9. Recall values of 100-word summary of proposed CatSumm + Eigenvector Centrality and other summarization methods using DUC-2004

4. Summary and Conclusions

In this research, we present CatSumm, a new unsupervised approach for summarizing multi-document text. The presented method consists of a series of steps aimed at establishing an order of importance among sentences. In the presented method, irregularities in everyday language are eliminated with a application called KUSH, which prepares texts for summarizing with an innovative approach. With this tool, successful results were obtained at the point of obtaining relations between sentences. In this study, graphs representing the summary were created by using techniques in algebraic graph theory known as spectral graph partitioning which contributed to the summarizing performance. Based on the results of the experimental studies, we believe that the developed KUSH software tool can be used prior to other classification and clustering methods frequently used by researchers in this field.

Finally, the proposed CatSumm approach was conducted on the basis of many nodes centrality methods and obtained a very high ROUGE value across all these methods. Comparisons were made with six summarization methods using the combination of the proposed CatSumm method and Eigenvector Centrality. Evaluations show that the best Recall values are obtained for abstracts of 100 and 200 words. In addition, the 400-word abstracts also showed very competitive results, with second-

best values clearly shown in the results tables and figures. These combined results demonstrate the robustness and stability of the proposed CatSumm text summarization method.

Contribution of the Authors

All authors contributed equally.

Conflict of Interest Statement

There is no conflict of interest between the authors.

Statement of Research and Publication Ethics

Research and publication ethics were complied with in the study.

References

[1] Durmaz O., 2011. Metin sınıflandırmada boyut azaltmanın etkisi ve özellik seçimi. 2011 IEEE 19th Signal Processing and Communications Applications Conference (SIU 2011) doi:10.1360/zd-2013-43-6-1064.

[2] Hark C., Uçkan T., Seyyarer E., Karci A. 2018. Metin Özetleme İçin Çizge Tabanlı Bir Öneri. IDAP 2018 - International Artificial Intelligence and Data Processing Symposium, 1–6.

[3] Canberk G. , Sağıroğlu Ş. 2006. Bilgi ve Bilgisayar Güvenliği : Casus Yazılımlar ve Korunma Yöntemleri (Grafiker Yayıncılık, Ankara).

[4] Uçkan T., Hark C., Seyyarer E., Karci A. 2019. Ağırlıklandırılmış çizgelerde Tf-Idf ve eigen ayrışımı kullanarak metin sınıflandırma. Bitlis Eren Üniversitesi Fen Bilim Derg. doi:10.17798/bitlisfen.531221.

[5] Hark C., Uckan T., Seyyarer E., Karci A. 2019. Extractive Text Summarization via Graph Entropy Çizge Entropi ile Çikarici Metin Özetleme. 2019 International Conference on Artificial Intelligence and Data Processing Symposium, IDAP 2019 doi:10.1109/IDAP.2019.8875936.

[6] Hark C., Seyyarer A., Uçkan T., Karci A. 2017. Doğal Dil İşleme Yaklaşimlari ile Yapisal Olmayan Dökümanlarin Benzerliği. IDAP 2017 - International Artificial Intelligence and Data Processing Symposium, 1–6.

[7] Radev DR., Hovy E., McKeown K. 2002. Introduction to the special issue on summarization. Comput Linguist, 28 (4): 399–408.

[8] Erkan G., Radev DR. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. J Artif Intell Res, 22: 457–479.

[9] Das D., Martins AFT. 2007. A survey on automatic text summarization. Lit Surv Lang Stat II course C, 4 (192–195): 57.

[10] Kaynar O., Görmez Y., Işık YE., Demirkoparan F. 2017. Comparison of Graph Based Document Summarization Method. 2017 International Conference on Computer Science and Engineering (UBMK), 598–603.

[11] Kutlu M., Cigir C., Cicekli I. 2010. Generic text summarization for Turkish. Comput J, 53 (8): 1315–1323.

[12] Alguliev RM., Aliguliyev RM., Hajirahimova MS. 2012. GenDocSum+ MCLR: Generic document summarization based on maximum coverage and less redundancy. Expert Syst Appl 39 (16): 12460–12473.

[13] Dalal V., Malik L. 2013. A Survey of Extractive and Abstractive Text Summarization Techniques. 2013 6th International Conference on Emerging Trends in Engineering and Technology (IEEE), 109–110.

[14] Hark C., Uçkan T., Seyyarer E., Karci A. 2019. Metin özetlemesi için düğüm merkezliklerine dayalı denetimsiz bir yaklaşım. Bitlis Eren Üniversitesi Fen Bilim Derg., doi:10.17798/bitlisfen.568883.

[15] Mihalcea R., Tarau P. 2005. A Language Independent Algorithm for Single and Multiple

Document Summarization. Proceedings of IJCNLP 2005, 2nd International Join Conference *on* Natural Language Processing, 19–24.
[16] Sarkar K., Saraf K., Ghosh A. 2015. Improving Graph Based Multidocument Text Summarization Using an Enhanced Sentence Similarity Measure. 2015 IEEE 2nd International Conference on Recent Trends in Information Systems, ReTIS 2015 - Proceedings, 359–365.
[17] Joshi A., Fidalgo E., Alegre E., Fernández-Robles L. 2019. SummCoder: An unsupervised framework for extractive text summarization based on deep auto-encoders. Expert Syst Appl 129: 200–215.
[18] Mihalcea R., Tarau P. 2004. TextRank: Bringing Order into Texts. Proceedings of the ACL 2004 on Interactive Poster and Demonstration Sessions - (Association for Computational Linguistics, Morristown, NJ, USA), 20.
[19] Parveen D., Ramsl H-M., Strube M. 2015. Topical Coherence for Graph-Based Extractive Summarization. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 1949–1954.
[20] Hark C., Karci A. 2020. Karcı summarization: A simple and effective approach for automatic text summarization using Karcı entropy. Inf Process Manag, 57 (3): 102187.
[21] Uçkan T., Karci A. 2019. Extractive multi-document text summarization based on graph independent sets. (xxxx). doi:10.1016/j.eij.2019.12.002.
[22] Luhn HP. 1958. The Automatic Creation of Literature Abstracts. IBM J Res Dev, 2 (2): 159–165.
[23] Edmundson HP. 1969. New methods in automatic extracting. J ACM, 16 (2): 264–285.
[24] Mallick C., Das AK., Dutta M., Das AK., Sarkar A. 2019. Graph-Based Text Summarization Using Modified TextRank. Soft Computing in Data Analytics (Springer), 137–146.
[25] Pouriyeh S., et al. Graph-based Ontology Summarization: A Survey.
[26] Allahyari M., et al. 2017. Text summarization techniques: A brief survey. doi:10.1145/nnnnnnn.nnnnnnn.
[27] Nasr Azadani M., Ghadiri N., Davoodijam E. 2018. Graph-based biomedical text summarization: An itemset mining and sentence clustering approach. J Biomed Inform, 84: 42–58.
[28] D'hondt J., Verhaegen P-A., Vertommen J., Cattrysse D., Duflou JR. 2011. Topic identification based on document coherence and spectral analysis. doi:10.1016/j.ins.2011.04.044.
[29] Uçkan T., Hark C., Karci A. 2020. SSC: Clustering of Turkish texts by spectral graph partitioning. J Polytech, doi:10.2339/politeknik.684558.
[30] Karci A. 1998. Çizge Algoritmaları ve Çizge Bölmeleme. Dissertation (Fırat Universitesi).
[31] Von Luxburg U. 2007. A Tutorial on Spectral Clustering.
[32] Slininger B. Fiedler's Theory of Spectral Graph Partitioning.
[33] Robert N.. Statistics: Definition of Standard Deviation.
[34] Bavelas A. 1948. A mathematical model for group structures. Hum Organ, 7 (3): 16–30.
[35] Fattah MA., Ren F. 2009. GA, MR, FFNN, PNN and GMM based models for automatic text summarization. Comput Speech Lang, 23 (1): 126–144.
[36] Boudin F., et al. 2013. A Comparison of Centrality Measures for Graph-Based Keyphrase Extraction To cite this version : HAL Id : hal-00850187 A Comparison of Centrality Measures for Graph-Based Keyphrase Extraction.
[37] Kosorukoff A. 2011. Social Network Analysis Theory and Applications (Passmore, D. L, 2011).
[38] Garey MR., Johnson DS. 1979. Computers and Intractability : A Guide to the Theory of NP-Completeness (W.H. Freeman).
[39] McPherson M., Smith-Lovin L., Cook JM. 2001. Birds of a feather: homophily in social networks. Annu Rev Sociol, 2 7(1): 415–444.
[40] Analysis BN. 2016. Centrality and Hubs. (1979). doi:10.1016/B978-0-12-407908-3.00005-4.
[41] NIST. Document Understanding Conferences. NIST.
[42] Lin CY. 2004. Rouge: A Package for Automatic Evaluation of Summaries. Proc Work text Summ branches out, (WAS 2004): 25–26.
[43] Lin C-Y., Hovy E. Automatic Evaluation of Summaries Using N-gram Co-Occurrence Statistics.

[44] Xiong S., Ji D. 2016. Query-focused multi-document summarization using hypergraph-based ranking. Inf Process Manag, 52 (4): 670–681.
[45] Republic C. 2009. Evaluation Measures for Text Summarization Josef Steinberger, Karel Je zek. 28: 1001–1025.
[46] Mihalcea R. 2005. Language Independent Extractive Summarization. Proc ACL 2005 Interact poster Demonstr Sess, - ACL '05 (June): 49–52.
[47] Mihalcea R., Tarau P. 1800. TextRank: Bringing Order into Texts.
[48] Vanderwende L., Suzuki H., Brockett C., Nenkova A. 2007. Beyond SumBasic: task-focused summarization with sentence simplification and lexical expansion. Inf Process Manag, 43 (6): 1606–1618.
[49] Haghighi A., Vanderwende L. 2009. Exploring Content Models for Multi-Document Summarization, (June): 362.