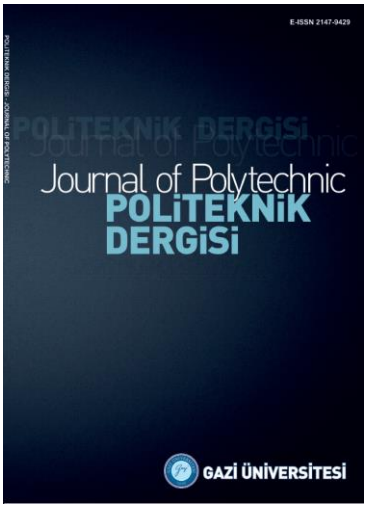

POLİTEKNİK DERGİSİ

JOURNAL of POLYTECHNIC

ISSN: 1302-0900 (PRINT), ISSN: 2147-9429 (ONLINE)
URL: http://dergipark.org.tr/politeknik

SSC: Clustering of Turkish texts by spectral graph partitioning

SSC: Türkçe metinlerin spektral çizge bölmeleme yöntemi ile kümelenmesi

***Yazar(lar) (Author(s))**: Taner UÇKAN[1], Cengiz HARK [2], Ali KARCI [3]*

ORCID[1]: 0000-0001-5385-6775

ORCID[2]: 0000-0002-5190-3504

ORCID[3]: 0000-0002-8489-8617

Bu makaleye şu şekilde atıfta bulunabilirsiniz(To cite to this article): Uçkan T., Hark C. ve Karcı A., "clustering of turkish texts by spectral graph partitioning", ***Politeknik Dergisi***, 24(4):1433-1444,(2021).

Erişim linki (To link to this article): http://dergipark.org.tr/politeknik/archive

DOI: **10.2339/politeknik.684558**

Ssc: Clustering Of Turkish Texts By Spectral Graph Partitioning

Highlights

- *KUSH text preprocessing tool*
- *Spectral graph partitioning*
- *Text clustering*

Graphical Abstract

In this study, first of all, raw texts are cleaned with KUSH tool. Then, the texts are converted to weighted graphs and clustering is done from the subgraphs obtained at the last stage.

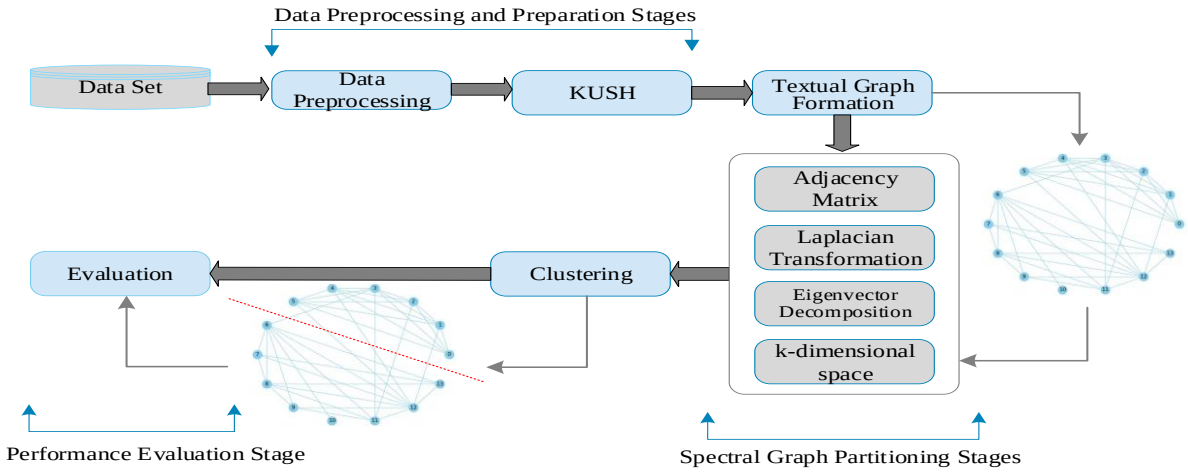

Figure. A schematic outline of the SSC model proposed for text clustering.

Aim

The main goal of this study is to present a new and consistent process text clustering by applying the spectral graph partitioning method to natural language data

Design & Methodology

The study explains how the proposed model proposed can be used in natural language applications to successfully cluster texts. A spectral graph theory method is used to partition the graph into non-intersecting sub-graphs, and an unsupervised and efficient solution is offered for the text clustering problem by providing a physical representation of the texts.

Originality

In this study, KUSH is used as a new tool as a Text Preprocessing step. In addition, text clustering is performed by using spectral graph division method.

Findings

As clearly stated in the related table, as a result of the promising results in terms of performance metrics used, it is seen that the proposed method can be a useful and effective tool in text clustering.

Conclusion

In terms of accuracy performance metric showing the overall effectiveness of the cluster, an average of 93.2% performance value was obtained.

Declaration of Ethical Standards

The authors of this article declare that the materials and methods used in this study do not require ethical committee permission and/or legal-special permission.

SSC: Türkçe Metinlerin Spektral Çizge Bölmeleme Yöntemi İle Kümelenmesi

Araştırma Makalesi / Research Article

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

[1]Başkale Meslek Yüksek Okulu, Bilgisayar Programcılığı Bölümü, Van Yüzüncü Yıl Üniversitesi, Türkiye
[2]Mühendislik Fakültesi, Yazılım Müh. Bölümü, Turgut ÖZAL Üniversitesi, Türkiye
[3]Mühendişlik Fakültesi, Bilgisayar Müh. Bölümü, İnönü Üniversitesi, Türkiye

(Geliş/Received : 05.02.2020 ; Kabul/Accepted : 05.06.2020 ; Erken Görünüm/Early View : 20.06.2020)

ÖZ

Mevcut veri miktarındaki artış nedeniyle metin kümeleme çalışmalarına olan ilgi artmaktadır. Metin kümeleme alanında farklı yaklaşımlar kullanılarak birçok çalışma yapılmıştır. Bu çalışmada, çizge bölmelemeye dayalı denetimsiz bir yöntem olan Spektral Cümle Kümelemesi (SSC) tanıtılmaktadır. Çalışma kapsamında, önerilen modelin metinleri başarıyla kümelemek için doğal dil uygulamalarında nasıl kullanılabileceği açıklanmaktadır. Metinlerin fiziksel temsili sağlandıktan sonra, metin kümeleme problemi için spektral çizge teorisi kavramları kullanılarak denetimsiz ve verimli bir çözüm sunulmuştur. Son olarak, SSC'nin metin kategorizasyonu amacı ile başarılı bir şekilde kullanılabileceğini gösteren testler yapılmıştır. Açık erişimli ve yapılandırılmamış Türkçe metinler içeren TTC-3600 veri seti kullanılarak yapılan testlerde %97.08'lik bir kümelenme başarı oranı elde edilmiştir. Önerilen SSC modeli, popüler k-ortalamalar kümeleme algoritmasına kıyasla daha iyi performans gösterdiği gözlemlenmiştir..

Anahtar Kelimeler: Çizge bölmeleme, spektral çizge teorisi, binary metin kümeleme, metin kategorizasyonu, metin madenciliği.

SSC: Clustering of Turkish Texts By Spectral Graph Partitioning

ABSTRACT

There is growing interest in studies on text classification as a result of the exponential increase in the amount of data available. Many studies have been conducted in the field of text clustering, using different approaches. This study introduces Spectral Sentence Clustering (SSC) for text clustering problems, which is an unsupervised method based on graph-partitioning. The study explains how the proposed model proposed can be used in natural language applications to successfully cluster texts. A spectral graph theory method is used to partition the graph into non-intersecting sub-graphs, and an unsupervised and efficient solution is offered for the text clustering problem by providing a physical representation of the texts. Finally, tests have been conducted demonstrating that SSC can be successfully used for text categorization. A clustering success rate of 97.08% was achieved in tests conducted using the TTC-3600 dataset, which contains open-access unstructured Turkish texts, classified into categories. The SSC model proposed performed better compared to a popular k-means clustering algorithm.

Keywords: Graph partitioning, spectral graph theory, binary text clustering, text categorization, text mining.

1. INTRODUCTION

Raw data are unassociated collections of information about a topic, and are also defined as transferable strings that can be expressed in the numerical format. Information, on the other hand, refers to raw data that are processed and brought together to form a meaningful whole [1]. The amount of available data, which is growing on a daily basis, leads to new problems awaiting solution. Raw texts without a fixed format make up a great majority of the digital data awaiting processing. These data need to be categorized and analyzed to become meaningful and useful sources of information. Rapid and easy access to information is only possible if data are categorized. The golden rule for accessing information with acceptable speed is to categorize data [1]–[3]. Working on large clusters of data imposes high calculation costs and requires intensive use of memory. The same transactions can be carried out at a much lower cost by categorizing or partitioning data.

As the internet becomes a bigger part of our daily lives and the amount of online information is multiplied by the day, there is growing demand for methods that can help organize high-volume documents [4], [5]. Partitioning can play a key role in organizing high-volume documents. Partitioning has been used for a long time in different fields including engineering, bioinformatics, medicine, and social sciences. Text clustering, which is a form of data clustering, is a data mining technique that utilizes concepts from the fields of information retrieval, natural language processing, and machine learning. Text clustering organizes documents into separate partitions called clusters. Each partition shares some common characteristics based on defined similarity criteria for text documents. Rapid and high-quality text clustering algorithms play an important role in helping users

*Sorumlu Yazar (Corresponding Author)
e-posta : taneruckan@yyu.edu.tr

convey, summarize, and organize information in an efficient manner [6], [7]. However, all text organization methods require selecting high-quality document collections, providing a good model, and identifying the appropriate techniques [8]. From an algorithmic perspective, text classification can be defined as the selection of relationships in large datasets. Text documents are collections with words as their terms. It is difficult to understand and interpret these text collections. Therefore, text based collections are transformed to usable formats by machines [9]. Even though there are hundreds of partitioning techniques, these techniques can be classified into a few main categories:

Disjoint clustering: This is a clustering method in which data are separated from one another with precise lines. In this clustering method, each member is assigned to a cluster in a clear and fixed manner. Clustering results in a group of distinct clusters. Clusters that form (C1, C2) are groups of data that do not intersect. Cluster information is clear, and members belong to one and only one of the existing clusters [10] [7].

Overlapping clustering: This is the clustering approach in which members are assigned to appropriate clusters in a fuzzy matter. Each member is allowed to be assigned to multiple clusters, creating a series of overlapping clusters [10] [7] [11].

Hierarchical clustering: This is a popular clustering approach. A new you partitioning is performed at every consecutive level, similar to a tree structure. This process starts with a single cluster that contains the entire data, and continues until in each cluster is represented by single piece of data. This process creates hierarchically nested clusters corresponding to the number of members making up the data set. These are distinct clusters [7], [10], [11].

Today, more and more researchers use the spectral graph theory to characterize the overall structure of graphs. The spectral graph theory aims to discover and reveal overall and structural characteristics of graphs using the eigenvalues and eigenvalue/vector pairs of the Laplacian matrix. There are many examples in the field of computer vision where spectral pairing methods are used for grouping and matching [12]. For example, [13] has shown that graphs with the same dimensions can be matched via singular value decomposition on adjacency matrices.

This study will use the graph partitioning method, which is commonly used in fields of study such as circuit design and map coloring [10]. Graph partitioning refers to grouping the nodes of a connected graph to form sub-graphs in such a way as to minimize the total segment weight on the graph [11]. A more detailed discussion of graph partitioning is provided in Section 2.

This study examines whether high partitioning performance values can be obtained by using algebraic graph theory techniques known as spectral graph partitioning to partition texts represented by graphs. To this end, tests of partitioning performance were conducted, which demonstrated that the SSC method can be used to partition text clusters with class labels that do not overlap. The performance of the SSC method was compared with the popular k-means clustering algorithm. To identify their performance levels, a series of metrics frequently used in text clustering were calculated using the TTC-3600 datasheet. The method has a sound and consistent infrastructure in that text sources are transformed into a physical structure via representation by graphs, and linear algebraic methods are used.

The following sections of the study are organized as follows. Section 2 presents the stages of the empirical analysis conducted, the dataset used, the SSC method developed, and the concepts. Section 3 explains the metrics and stages of the empirical analysis. Section 4 presents and discusses empirical results regarding the text clustering method we have proposed.

2. MATERIAL and METHOD

Figure 1 provides the block diagram stages for text clustering carried out using the graph partitioning method. The TTC-3600 dataset is a new, open-access dataset consisting of Turkish texts categorized using six different labels [12]. This is unstructured data. Unintegrated data that need to be processed and categorized must first undergo some preparatory steps. These steps were not limited to the removal of non-discriminating data from the system. Following the stage of data pre-processing and cleansing, the software tool KUSH[13] used as part of this study was used to prevent words that have similar or very similar meanings from being perceived as having different meanings -because of the suffixes they receive- when creating the graphs. Following the modification of the words making up the texts using the software tool KUSH, texts were represented as undirected and weighted graphs. The first of the finite sets (nodes) making up the graph are represented by sentences, and the second finite set (edges) is represented by common word counts. Prior to the stage of graph partitioning, the texts were cleansed and normalized, and graphs were obtained to represent the sentences. In the last stage, graphs are divided into subgraphs, maximizing the differences between them and minimizing similarities. At this stage, the SSC method was used for graph partitioning, which consists of steps explained in detail in section 2.4.

2.1. Dataset

This study uses the TTC-3600 dataset, which can be used for purposes of Turkish natural language processing, machine learning, text classification, text clustering, and the like. TTC-3600 is a new dataset consisting of Turkish language news articles and texts for studies involving Turkish texts. The most important feature of this dataset

is that it is easy to use and well documented. The dataset consists of a total of 3600 documents classified into 6 categories -economics, culture and arts, health, politics, sports, and technology-, each containing 600 texts obtained from well-known news portals and news agencies. There are other datasets used in different studies, but many of them are inaccessible or were created for different purposes. TTC-3600 is a new and open-access dataset [12]. For these reasons, this dataset was preferred when using KUSH, the software tool used, and measuring the categorization performance of the SSC method proposed..

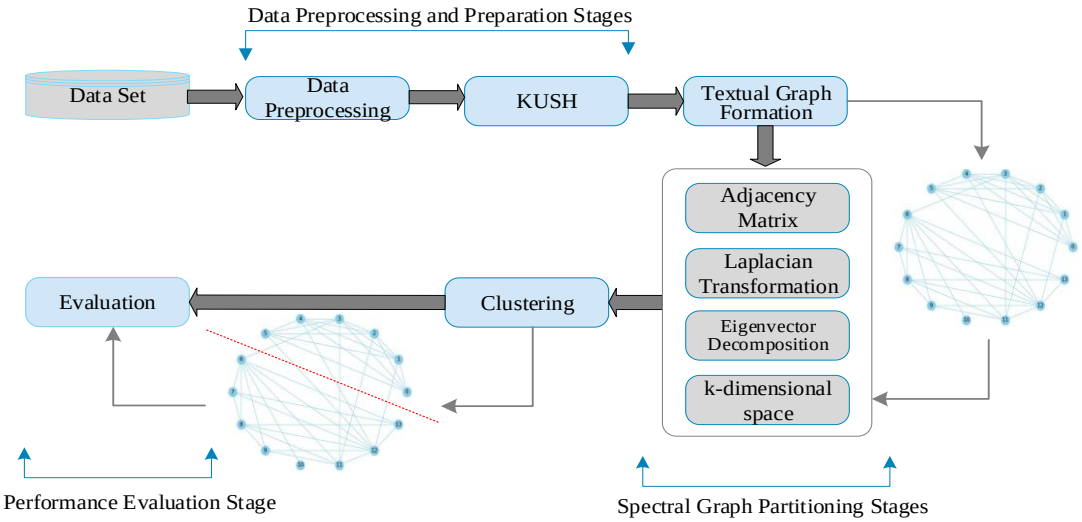

Figure 1. A schematic outline of the SSC model proposed for text clustering.

2.2 Data Pre-Processing and Preparation

Most data sets are unstructured; only very few of the relevant data sets have structured formats. Structured data sets can be expressed in the form of rows and columns in a table or have tags. In this study, structuring the data to be processed was only possible after a complicated and time-consuming process. Unintegrated datasets that need to be structured first have to undergo some pre-processing steps. It is predicted that transforming the texts to be used as inputs in the system to usable formats and removing unnecessary, non-discriminating data would improve the performance of the system, because the datasets in question have been created using natural language [14]. In text classification, data is not used without pre-processing. Stop-words (pronouns, prepositions, conjunctions, etc.) are non-discriminating phrases that need to be removed from the dataset prior to categorization. Therefore, for data to be converted into usable formats and text clustering to be performed, stop-words that have no representative power for sentences are removed from the original dataset. This ensures that the transactional load for the process is reduced and only words with categorical meaning are retained [2], [12]. In the TTC-3600 dataset, texts to be represented with graphs are stored in files with the .txt extension. In this study, normalization steps, that is to say the removal of stop-words, spaces, and unwanted characters, were carried out using Python libraries and tags.Also in this study, the kush tool was used in the text pre-processing stage.

These normalization steps make-up the first part of the data pre-processing and preparation stage. In addition, when related words that have the same root but are written differently are treated as different words, this makes it more difficult to identify the connections and relations between sentences. It was intuitively predicted that this problem would have a significant effect on the categorization performance, and a text processing tool was developed for use prior to creating text graphs within the framework of the SSC model. This software tool named KUSH was developed using C# on the .NET platform. The pseudo code of our KUSH is shown in Algorithm 1.

Changes in words are made taking into account the n-gram intersections and the relationship with other words in the text. In short, a different approach has been proposed in addition to some well-known Text Normalization methods. In the proposed approach, the words in the main text are converted to their simplest forms. Due to all these differences, the proposed KUSH text processing tool differs from the standard NLP preprocessing tools. The aim was to provide maximum representation in the text presentation step.

This basic transformation is not made considering the meaning of the related word in the language it belongs to. The word is translated into the simplest form based on n-gram in the text to be clustered. In this study, synonyms, contrasts etc. were not considered. Within the scope of the study that text clustering based on spectral analysis.

Therefore, the interpretation and re-expression of sentences is beyond the scope of our study.

- ✓ Concepts,
- ✓ Terms,
- ✓ Sentences,

Algorithm 1. Proposed KUSH algorithm

Input: {text} - texts in everyday language
Output: {kush_text} - texts processed by KUSH algorithm

```
1   "text read";
2   sentences=[], words=[], alternative =[];
3   best_alternative =0, kush_text={}
4   sentences []=text.Split(.);
5   for c to len(sentences) do
6       sentences [c]=clear(c);
7   end for
8   words[]=sentences.Split(.);
9   for k to boyut(words) do
10      alternative [k]= alternative_search(words [k], words);
11      for u to alternative do
12          best_alternative = best_alternative_search (k, alternative);
13      end for
14      words [k]= best_alternative;
15  end for
16  for k to words do
17      kush_text+=k+" ";
18  end for
19  return kush_text;
```

2.3 Text Graphs

Şekil Problems can be modelled graphically via formal representation of features and relationships between these features. In conceptual terms, graphs consist of nodes, and edges that represent the relationships between nodes. Nodes and edges are two finite sets. Nodes are the main members of the collection represented by the graph. Edges, on the other hand, are the relationships between the main members. In general terms, graphs are expressed as $G=(V,E)$. The set of nodes is $V=\{v_1, v_2,...,v_n\}$, *and the set of edges is* $E=\{e_1,e_2,...e_n\}$ $(E\subseteq VxV)$. For the edge $e_i=\{v_j,v_{j+1}\}$ end nodes are the nodes v_j and v_{j+1}. If $e_{1=}(v_j,v_{j+1})\in E$ and $(v_{j+1},v_j)\in E$ for neighboring nodes v_j *and* v_{j+1} , these edges are undirected edges. Graphs consisting of such edges are undirected graphs. If $e_{1=}(v_j,v_{j+1})\in E$ and $(v_{j+1},v_j)\notin E$ for neighboring nodes v_j and v_{j+1}, these edges are directed edges. Graphs consisting of directed edges are called directed graphs [15], [16]. If the edges $E=\{e_1,e_2,...e_n\}$ representing the relationships between the nodes $V=\{v_1, v_2,...,v_n\}$ on a graph $G=(V,E)$ have non-negative weights, such graphs are called weighted graphs [17]. The graphs created in this study to represent the texts are undirected and weighted graphs.

There are different approaches in the literature to the representation of text graphs. Full sentences or quasi-sentences can be represented by nodes, whereas different types of relationships between nodes, such as intersection and concurrence, can be represented by edges.

In general terms, nodes on a graph representing texts may correspond to

- ✓ Paragraphs, or
- ✓ Documents.

To represent the relationships between nodes, on the other hand, different representation types with different approaches can be used, including

- ✓ Semantic relationships,
- ✓ Word commonalities, and
- ✓ Framework model (frequency of occurrence in strings of a certain size).

This study, graphs were created to represent texts. Nodes, making up the first finite set in these representative graphs, correspond to sentences, whereas edges, making up the second finite set, correspond to the number of common words. Thus, texts were represented using undirected and weighted graphs. In terms of expression, there are different representation types such as

- ✓ Dynamic expression and
- ✓ Matrix expression.

Different types of expression have their own advantages and disadvantages. As Karcı notes in [16], dynamic expression of graphs provides advantages in terms of memory use, and allows intervening in the number of nodes and edges during operation. Despite these advantages, however, it creates various problems in terms of usage. Memory space is allocated for every node and edge. In this method, a set of nodes is defined in the form of a list of neighboring nodes or edges. Linear connected lists are used as the neighboring set of any node. These factors make the dynamic approach

advantageous in terms of memory use. However, dynamic approaches are very complex to use.

For a graph *G=(V,E)*, the adjacency matrix *A(G)=($a_{i,j}$)* where the node labels of the graph are used as matrix indices is a *vxv* symmetric matrix and is defined by Equation (1). Here, $W_{i,j}$ is the weight of the edge connecting two nodes.

$$a_{i,j} = \begin{cases} w_{i,j} & , (i,j) \in E \\ 0 & , (i,j) \notin E \end{cases} \quad (1)$$

The degree matrix D(G)=(di,j) for a graph G=(V,E) is a vxv diagonal matrix, and is defined by Equation (2).

$$d_{i,j} = \begin{cases} d_{i,j} & , i = j \\ 0 & , i \neq j \end{cases} \quad (2)$$

This study prefers to use matrix representation to express graphs. After carrying out the data pre-processing and preparation steps explained in Section 2.2, the proposed SSC model is used to create graphs representing texts so that spectral graph partitioning methods can be applied to textual data. Operations carried out in this section correspond to the stage of Creating Text Graphs shown in Figure 1. Figure 2 contains a simple text sample and the Sentence-Word Graph created for this text. The Sentence-Word Graph is a weighted and undirected graph. In creating the Sentence-Word Graph, we are proposing a transformation where nodes represent sentences and edges represent the number of common words. For each sentence, the number of common words the sentence has with all other sentences is calculated, and the strength of the relationship in individual sentence pairs is established. This makes it possible to express sentences and relationships between sentences using a highly representative graph. Sentence-Word Graph is a graph that reveals the strength of the connections and relationships between all sentence pairs in the text to be categorized.

For each sentence in the texts, a node was added to the representative graph. Edge weights were added to the edges between nodes on the basis of the number of common words between sentences. If there were no common words between two sentences, no edge was added between the relevant nodes. Figure 2 provides a sample paragraph after the data pre-processing and preparation steps, and the graph that corresponds to this paragraph. The sample paragraph consists of two separate texts from the Health and Technology categories of the TTC-3600 data set. The first six sentences belong to the Technology category, and the remaining eight sentences belong to the Health category.

2.4 Graph Partitıoning

Graph partitioning is the general name for methods that aim to find an optimal solution maximizing intra-cluster similarities and minimizing inter-cluster similarities. Graph partitioning is an important problem with applications in fields such as circuit design, load balancing, optimization, and parallel calculations. Scientific calculations on data represented by graphs involve problems from different fields, including algebra, probability theory, ergodic theory, and geometry. In these types of problems, problem space needs to be divided into distinct subsets to increase the efficiency of the solution and make the process more manageable. To this end, the graph representing the data needs to be partitioned in the least costly manner possible, as shown in Figure 3. The graph is created on the basis of the data constituting the problem, and partitioned using different methods. The graph partitioning problem has an exponential mathematical complexity, that is to say, it is an NP-hard problem. Even though there are different graph partitioning algorithms, none of these algorithms can guarantee the ideal solution. However, many of the graph partitioning methods presented generate good results [11], [16], [18], [19].

(0) Amazon'dan yapılan açıklamada, e-kitap okuyucusu Kindle'ın sınırsız üyelik seçeneğine abone olanların okuduğu sayfa sayısının kitaptan elde edilen geliri belirleyeceği ifade ediliyor. **(1)**Şirket daha önce indirilen kitap adedi başına telif ödemesi yapıyordu. **(2)**Uzmanlar yeni uygulamayla birlikte daha uzun kitaplar yazmanın daha kârlı hale gelebileceğini belirtiyor. **(3)**Amazon yeni ödeme sistemiyle birlikte kitapların başarısının daha sağlıklı takip edilebileceğini vurguladı ve yeni uygulamaya geçme kararının bağımsız yazarlarla uzun süreli görüşmelerin ardından alındığını ifade etti. **(4)**

…

...

(10) Araştırmanın sonuçları "Contact Lens & Anterior Eye" dergisinde yayımlandı. **(11)**Bilgisayar başında uzun süre oturmak göz sağlığını tehdit edebilecek pek çok etkenin bir araya gelmesine ve kalıcı rahatsızlıklara yol açabiliyor. **(12)**SOMON, TON BALIĞI, SEMİZOTUNDA OMEGA 3 VAR ….. zorluğu, gözleri kısarak bakma, ışığa karşı duyarlılık, göz kapaklarında iltihap, baş ağrısı gibi sorunlara neden olabiliyor. **(13)**Omega 3, ton, somon gibi balıkların yanı sıra keten tohumu, semizotu, karanfil, ıspanak, soya fasulyesi gibi besinlerde de bolca bulunuyor.

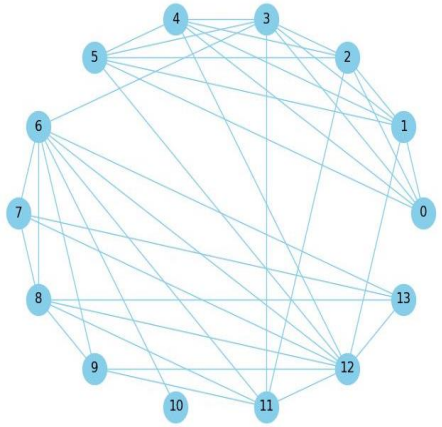

Figure 2. Sentence Word Graph for two sample documents from the Technology and Health categories of the TTC-3600 dataset after transformation with the Kush tool.

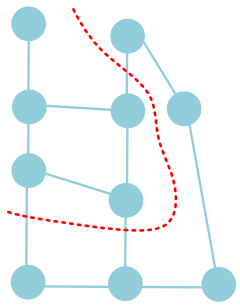

Figure 3. Symbolic representation of graph partitioning

The idea of spectral clustering utilizes methods of the spectral graph partitioning theory. Donath and Hoffman have argued that eigenvalue vectors of similarity matrices were related to graph partitioning problems. Fiedler, on the other hand, has proved that graph partitioning was closely related to the second smallest eigenvalue vector of the Laplacian matrix[19]. Similarly, according to the Rayleigh-Ritz theory discussed in [20], eigenvalue vectors of the Laplacian matrix can be used to approximate the optimal solution for graph partitioning.

An undirected and weighted graph $G=(V,E)$ is created to represent a data set or collection. $V=\{v_1, v_2,...,v_n\}$,is the set of nodes, and $E=\{e_1,e_2,...e_n\}$ is the set of weighted edges between nodes. The graph partitioning problem for graph G is defined by Equation (3).

$$cut(A,B) = \sum_{i\in A, j\in B} w_{ij}\,, \bar{A} = \{V_p \mid V_p \in V \; ve \; v_p \notin A_i\} \quad (3)$$

$$cut(A_1 \dots A_k) = \sum_{i=1}^{k} cut(A_i, \bar{A}_i) \quad (4)$$

A good graph partitioning process minimizes the objective function. However, this objective function does not take the density distribution in the clusters into count, and is based on the external connections of clusters only. This may create unbalanced data clusters due to the presence of outlying or unbalanced data in the dataset. In [21], Hagen and Kahng proposed normalized cut to obtain more balanced data clusters. Normalized cut proposes adding the cluster size ((A_i)¯) as a balancing variable to the cut function to minimize similarities between the clusters to be obtained from the objective function. The goal is to prevent unbalanced clustering.

$$Normalized\ Cut(A_1 \dots A_k) = \sum_{i=1}^{k} \frac{cut(A_i, \bar{A}_i)}{\bar{A}_i} \quad (5)$$

However, fractional cut is not sufficient to obtain a better performance in terms of cluster similarities. Therefore, Shi and Malik have proposed normalized cut as defined in Equation (6). Minimizing the objective function means minimizing the total weights between the subgraphs and simultaneously maximizing the internal weights of each subgraph; therefore, this approach produces better classification results [19], [21], [22].

$$vol(A_i) = \sum_{p\in A, q\in V} w_{pq}\,, Ncut(A_1 \dots A_k) = \sum_{i=1}^{k} \frac{cut(A_i, \bar{A}_i)}{vol(A_i)} \quad (6)$$

This study uses the spectral graph partitioning method; graph partitioning is achieved by obtaining graph eigenvalues and the eigenvalue vectors corresponding to these eigenvalues.

2.4.1 Spectral Graph Partitioning

In recent years, there is a growing interest in the use of spectral graph theory to characterize the overall structure of graphs. The spectral graph theory aims to summarize overall and structural characteristics of graphs using the eigenvalue vectors of adjacency matrices or Laplacian matrices. Spectral methods constitute an important part of many numerical algorithms. Spectral methods are commonly used in matrix partitioning, graph partitioning, and circuit simulations [11], [16]. As Figure 4 shows, spectral clustering algorithms are able to group objects with irregular shapes in terms of connectivity, differently from k-means and other clustering algorithms.

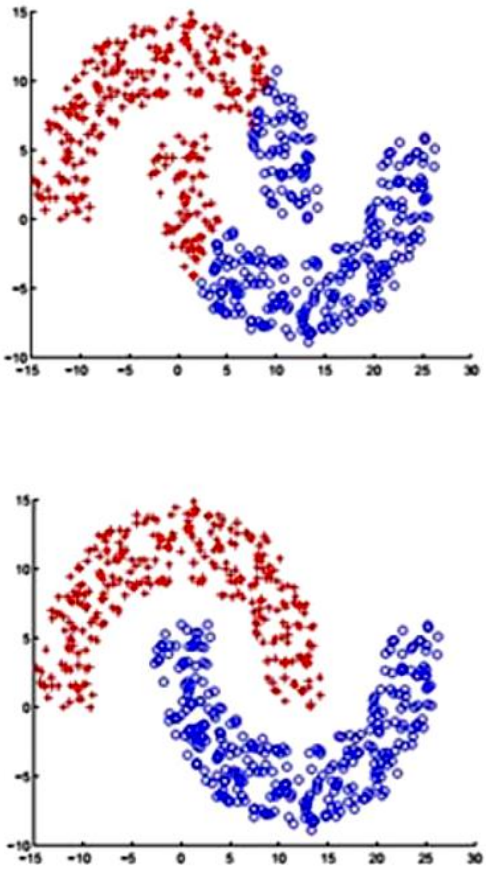

Figure 4. Comparing k-means and spectral clustering methods (adapted from [23])

Deneylerde Spectral clustering uses the concepts of spectral graph theory. The clustering problem optimizes a suitable objective function and structures it as a graph cut problem. Figure 5 shows the steps of the spectral

clustering algorithm. The main idea is to construct a graph from the dataset prepared. Each node in the dataset represents a relationship, and each weighted edge simply takes into account the similarity between two relationships. A clustering problem with this structure can be viewed as a graph cut problem, and this is known as the spectral graph theory. This theory is based on the decomposition of the eigenvalues and eigenvalue vectors of the Laplacian matrix of the weighted graph obtained from the data [24]. There are many Laplacian calculation methods in the literature, such as simple Laplacian, normalized Laplacian, and generalized Laplacian. All of these Laplacian methods use the diagonal degree matrix, which measures the degrees on nodes. To balance the relationship matrix with respect to clusters, the degree matrix is used as a normalizing factor [25], [26]. The Laplace matrix for any graph is a real and symmetric matrix. In the method firs proposed by Fiedler in [27], eigenvalue vectors corresponding to the second smallest eigenvalue of the matrix can be taken as the algebraic connectivity of the matrix, and used to divide the graph into two parts. The second smallest eigenvalue of the matrix is called the Fiedler value because of Fiedler's proposal. The eigenvalue vector holding the minimum separator of the graph is called the Fiedler vector.

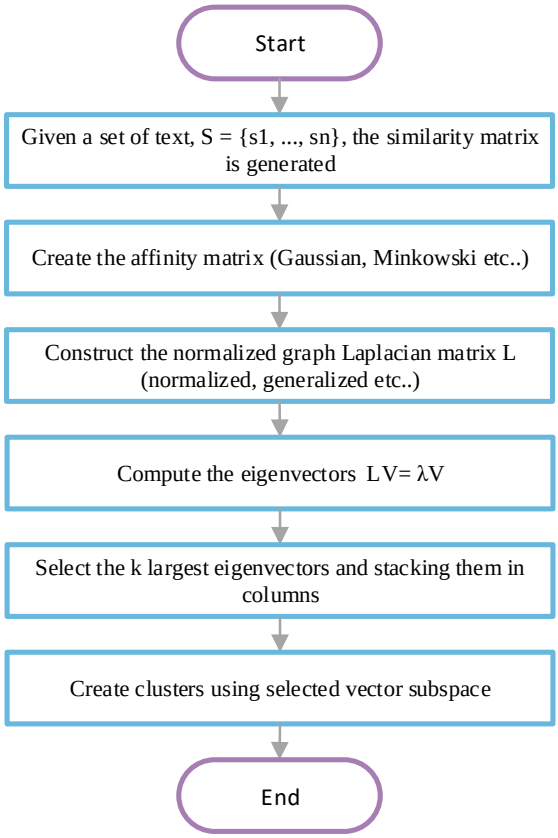

Figure 5. Spectral Clustering Algorithm

In a graph *G=(V,E)*, *V* is the set of nodes, and *E* is the set of edges. The *nxn* adjacency matrix of the graph *G* for */V/=n* is expressed by Equation (1). The *nxn* degree matrix of the graph *G* is calculated as in Equation (2). The degree matrix is a diagonal matrix. Of the Laplacian methods introduced in Section 2.5, this study uses the simple Laplacian. Simple Laplace is calculated as in Equation (7) using the adjacency matrix and the diagonal degree matrix of the graph, and the separate components of the Laplace matrix is calculated as in Equation (8).

$$L = D - A \quad , \quad d_i = \sum_{\{j|(i,j)\in E\}} w_{ij} \tag{7}$$

$$LaplaceG(i,j) = \begin{cases} \sum_{(i,k)\in E} A(i,k) & , \quad if\ i = j \\ -A(i,j) & , \quad if\ i \neq j \\ 0 & , \quad other \end{cases} \tag{8}$$

The Laplace matrix has a number of important characteristics. It is a positive semidefinite matrix that is also symmetrical. Eigenvalues and the corresponding eigenvalue vectors are obtained from the Laplace matrix. The graph spectrum for graph *G* is obtained by arranging the eigenvalues in descending order. Graph spectrum also contains important information about the connectivity of the graph. The eigenvalues for *L(G)*, the Laplace matrix of the graph *G*, are r ($0 = \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$). The set of eigenvalues of *L(G)* is usually called the spectrum of *L(G)* or the spectrum of the graph *G* [11], [16], [28]. The spectrum of *L(G)* also contains information about the connection paths of the graph G. Eigenvalues can be used to reveal overall graph characteristics that may not be clear in edge structures. For example, if the eigenvalues of *L(G)* for k different eigenvalues are calculated to be zero $0 = \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_k$, then graph G consists of *k* connected components. This example of graph connectivity is shown in Figure 6. Examining the spectrum ($0 = \lambda_1 = \lambda_2$, $\lambda_3 > 0$) of the $L(G_1)$, eigenvalues λ_1 and λ_2 are equal to zero, and as predicted by the theory [27], the graph clearly consists of two parts. In the spectrums of $L(G_2)$ and $L(G_3)$, on the other hand, only the eigenvalue λ_1 is equal to zero, and it can be seen that all the nodes in graphs G_2 and G_3 are connected, and these graphs have only one part. Graph G_3 has higher connectivity than graph G_2. This can also be predicted by comparing their λ_2 values, using the Fiedler theory $L(G3)(\lambda_2) > L(G2)(\lambda_2)$.

2.4.2 Laplace Matrix and Fiedler Vector

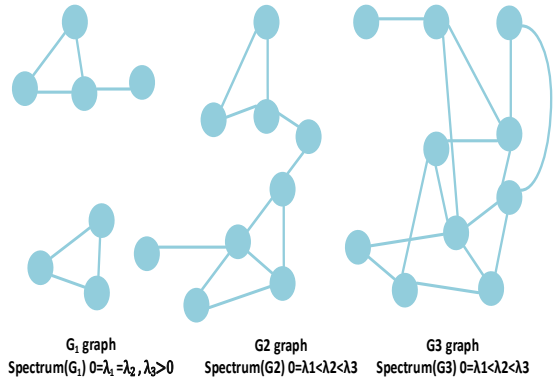

Figure 6. The relationship between graph spectrum and graph connectivity

2.4.3 SSC: Spectral Sentence Clustering Algorithm

As mentioned above, the nodes of the Sentence-Word Graph represent the sentences in the texts. The SSC method we have proposed divides the nodes of the graph created into groups. This approach associates each sentence making up the main graph with other sentences. Sentences are clustered using these relationships. The software tool group aims to provide these clusters with a more accurate and salient structure. We are using techniques of the spectral graph theory, which helps analyze the overall structural characteristics of a graph and is also used in other fields. These techniques aim to divide graph nodes into non-intersecting groups that are as equivalent as possible with the smallest cut cost, by taking into account the weights or distances between nodes (minimizing or maximizing them). In the method proposed, we are using techniques of the spectral graph theory to cluster sentences from texts categorized by subject. The pseudo code of our clustering method is shown in Algorithm 2. The main idea in Algorithm 2 is to group nodes corresponding to sentences in text documents based on their commonalities with one another. This, in turn, was based on the prediction that sentences in the same category would be similar to each other, and sentences in different categories would have fewer similarities with one another.

The way the SSC approach works is demonstrated using a sample text. In Figure 7, nodes belonging to the representative clusters obtained are shown with different colors. Sentences grouped by the SSC model in the same category are more closely associated with one another than with sentences in different categories.

Algorithm 2. SSC: Spectral Sentence Clustering Algorithm

Input: Texts processed by KUSH algorithm
Output: { C_1, C_2 } | $C_1 \cap C_2 = \emptyset$ - Discrete clusters of texts

```
degree_matrix=[][], adjacency_matrix =[][] , laplace_matrix=[][];
relationship_value=0, degree_value=0, eigen_value =0, eigen_value _vec=0
C1={}, C2={}, ssc_cluster_list={}
for i to len(text) do
  for j to len(text) do
    if (sentence(i) ∩ sentence(j)) > 0 and i!=j then
      relationship_value= len(sentence(i),sentence(j))
          degree_value+=1
        else
          relationship_value=0
        end if
      end for
      adjacency matrix[i,j]=relationship_value, degree_matrix[i,j]=degree_value
end for
laplace_matrix = (degree_matrix - adjacency matrix)
eigen_value, eigen_value_vec = eigen_decomposition (laplace_matrix)
eigen_value = sort(eigen_value)
for i to len(eigen_value _vec) do
      if eigen_value _vec[i,2] > 0 then
        C1.add(i), ssc_cluster_list.add(0)
      Else
        C2.add(i), ssc_cluster_list.add(1)
      end if
end for
return C1,C2, ssc_cluster_list
```

3. EMPIRICAL ANALYSIS

The main goal of this study is to present a new and consistent process text clustering by applying the spectral graph partitioning method to natural language data. In this context, empirical analysis was carried out to examine and demonstrate the effectiveness of the proposed SSC by classifying the texts in the TTC-3600 dataset. In addition, a software tool called KUSH was used as part of the study to improve the performance of the SSC method, and texts were pre-processed using various semantic processes. The success of the partitioning obtained was measured by calculating various performance metrics frequently used in the literature, and the performance of the approach was compared with the performance of the conventional k-means clustering algorithm.

Steps of the empirical analysis carried out are shown in Figure 1. To cluster the texts, the texts were first cleansed of stop-words, and undesired and non-representative phrases and characters were removed from the dataset. Data preprocessing for the empirical analysis consisted of important and highly time-consuming steps. Python libraries and tags were used to pre-process the texts in the dataset, which were categorized by subject and stored in .txt format. As Figure 1 shows, a text processing tool named KUSH was used and used after the data pre-processing stage and before creating text graphs. Because

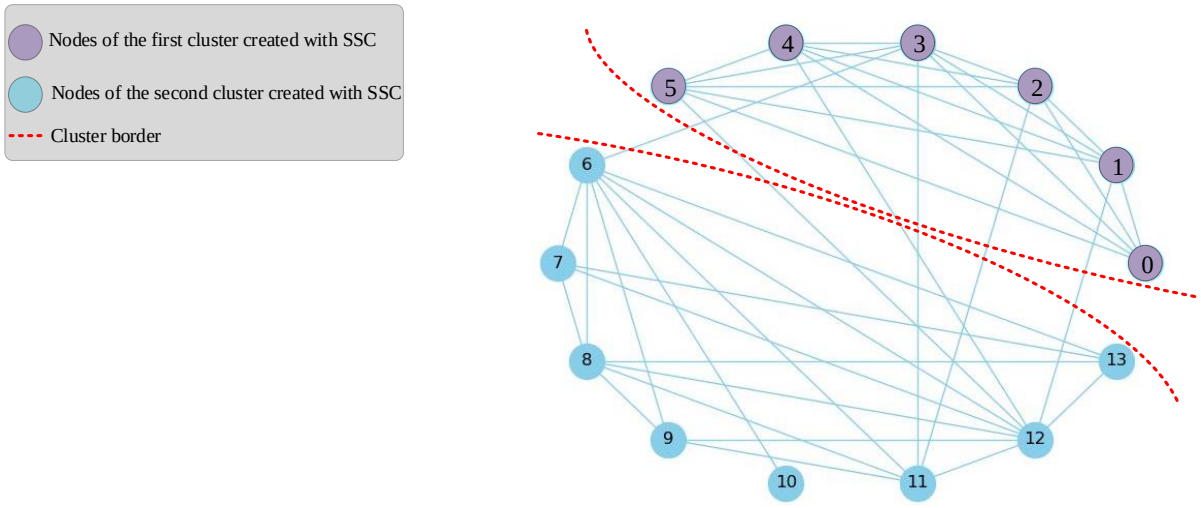

Figure 7. Partitioning the sample Sentence-Word Graph in Figure 2 using the SSC model proposed.

spectral methods have exponential mathematical complexity, the best solution cannot be guaranteed but they usually produce good results. The text processing software KUSH contributes the success and robustness of the spectral method. Text classification carried out without using the software KUSH failed to obtain high performance values. Therefore, the software used constitutes an important part of the SSC approach. In the next stage, Python tags were used to create graphs where nodes represent the sentences and edges represent the number of common words, and to create relationship matrices for these graphs. The degree matrix was used as a normalizing factor to balance relationship matrices. There are different Laplacian calculation methods in literature, and the simple Laplacian calculation given in Equation 7 was used in the present study. The Laplacian matrices created were used to carry out eigenvalue and eigenvalue vector decompositions, and the eigenvalues and eigenvalue vector pairs obtained were sorted. This sorting has produced the spectrum of the text graph, which contains important information about the connectivity of the graph. Eigenvalue vectors corresponding to the second smallest eigenvalue are considered to be the algebraic connectivity of the graph. This vector was used to partition the graph in the empirical analysis. Thus, in the empirical analysis, graphs obtained from natural language documents were partitioned using an unsupervised method. SSC processes have identified the natural language categories of the text documents, and made it possible to identify subject categories in a given text document. In the SSC method presented in this study, only the stage involving the software KUSH has a semantic infrastructure. In the text processing software KUSH, which improved categorization performance, a semantic process was used to identify words that have the same meaning or similar meanings. The method has a very flexible infrastructure. It can be adapted to different languages. Throughout the empirical analysis, no user-defined classification parameters were needed or used.

To evaluate the clustering performance of the method, text categories assigned by the proposed method were compared with the actual text categories in the dataset. To this end, clustering performance was measured using some evaluation metrics. These metrics are accuracy (ACC), normalized mutual information (NMI), f-score and Adjusted Rand Index (ARI). These performance measurement methods are commonly used in text clustering. Accuracy calculates the number or percentage of correct predictions. Acc has a maximum value of one, denoting the case in which all the predicted clusters perfectly match the labeled clusters, and a minimum value of zero, denoting no match. If âi is the predicted value of the ith sample, and ai is the corresponding correct value, the ratio of correct predictions over n samples is expressed by Equation (9) [7], [29]–[31].

$$Acc(a_i, \hat{a}_i) = \frac{1}{n}\sum_{i=0}^{n-1} 1\,(\hat{a}_i = a_i) \qquad (9)$$

Normalized mutual information is a popular metric used to evaluate the division between T set tags and C set. By normalizing the mutual information (MI) value, normalized mutual information is calculated that takes a value between one representing the maximum match and zero representing the minimum match. This metric is independent of the exact values of the tags. Changing the class or cluster tag values in no way changes the metric value. NMI is calculated as in Equation (10).

$$NMI(T, C)\,\frac{MI(T,C)}{\sqrt{H(T)H(C)}} \qquad (10)$$

The MI(T,C) value in the equation refers to the MI value between T and C. H(T) and H(C) are entropies. The expression in the denominator is used to normalize the MI value in the interval [0,1].

Another metric we have used to test the performance of the clustering we have carried out is the F-score evaluation criterion. The F-score is an evaluation criterion calculated by matching each cluster with a class. Its calculation involves precision and recall.

$$F\_measure(C) = \sum_{c_i \epsilon c} \frac{C_i)}{S} max_{K_j \epsilon K}\{F(C_i, K_j)\} \quad (11)$$

$$F(C_i, K_j) = \frac{2\ x\ Recall(C_i, K_j) x\ Precission(C_i, K_j)}{Recall(C_i, K_j) x\ Precission(C_i, K_j)} \quad (12)$$

$$Recall(C_i, K_j) = \frac{n_{ij}}{C_i}\ , Precision(C_i, K_j)\ \frac{n_{ij}}{K_i} \quad (13)$$

nij is the number of members of the class i in cluster j. Kj is the number of members in cluster j. Ci is the number of members of class i.

Adjusted Rand Index (ARI) is the last metric we have used to confirm the performance of the clustering we have carried out. It is an adjusted version of the Rand Index (RI). Let G and C represent, respectively, true class information, and C represent the calculated cluster. In this case, RI (C, G) is calculated as shown in Equation (14).

$$R(C, G) = \frac{Tp+Tn}{Tp+Tn+Fp+Fn} \quad (14)$$

The value of RI varies between 0 and 1. A value of 1 indicates perfect match. RI does not take the natural structures of clusters into account. This problem is solved by using the ARI index [32].

$$ARI(C, G) = \frac{RI(C,G)-E(RI(C_R,G))}{max\ RI(C,G)-E(RI(C_R,G))} \quad (15)$$

Here, E(RI(CR,G)) is the expected value of RI [33].

In the empirical analysis, accuracy, NMI, F-score and ARI performance metrics were calculated to evaluate the text clustering success of the proposed model. The same metrics have also been calculated for k-means, which is a conventional clustering algorithm. The same data pre-processing and preparation steps were followed for both the SSC model proposed and the conventional k-means algorithm. Table 1 summarizes the clustering performances of our method and the k-means method.

As Table 1 clearly shows, accuracy and F-score values of the SSC model for different category pairs are very close. The same table also shows that there is more variation in the NMI and ARI performance metrics compared to accuracy and F-score values.

The highest accuracy, NMI, f-score, and ARI performance metrics for the SSC model (97.08%, 81.12%, 97.00%, and 88.60%, respectively) were calculated for the Sports and Technology category pair. The highest values for the k-means algorithm (98.01%, 88.10%, 98.00%, 92.30%) were also calculated for the category pair Sports and Technology.

The mean accuracy value for the proposed model across all category pairs was 93.2%. For the k-means algorithm, the mean value was 89.00%. The mean value of the NMI, another metric frequently used to evaluate text clustering performance, was 67.52% for the SSC model, and 56.58% for the k-means algorithm. The mean f-score value was 93.32% for the SSC model, and 89.13% for its

Table 1. Performance metrics for clustering of the texts in the TTC-3600 dataset.

Pairs of Text Categories	**Clustering Performance Metrics (%)**							
	Accuracy		NMI		F-score		ARI	
	SSC	k-means	SSC	k-means	SSC	k-means	SSC	k-means
Economy, Culture-Arts	89.76	87.40	55.00	56.00	91.30	88.70	62.90	55.60
Culture-Arts, Health	93.60	78.60	65.10	29.40	95.10	78.70	75.90	32.10
Health, Politics	95.55	86.70	78.00	46.80	95.2	88.00	82.80	53.30
Politics, Sports	92.85	93.90	63.00	68.10	92.30	92.50	73.20	76.80
Sports, Technology	97.08	98.01	81.12	88.10	97.00	98.00	88.60	92.30
Technology-Economy	90.04	89.40	62.90	51.10	89.02	88.90	65.00	61.60
Average	93.2	89.00	67.52	56.58	93.32	89.13	74.73	61.95

competition. Finally, the mean value of the ARI index was 74.73% for SSC, and 61.95% for the conventional k-means algorithm, lower than the proposed model. In short, the performance of the proposed system was tested using metrics adopting different principles, and it was found to be superior compared to the alternative clustering algorithm..

4. RESULTS AND RECOMMENDATIONS

Using spectral clustering technique, this study has presented a graph-based approach called SSC for clustering natural language texts. Our method is based on relationships among eigenvalue vectors.

Disadvantages of the proposed clustering approach, namely the uncertainty of spectral methods and their inability to guarantee the best solution, were removed by the software tool named KUSH, which prepares texts for SSC using its unique algorithm. In this study, the spectral graph partitioning method was adapted and used for text classification. The mathematical complexity of the solutions offered in the literature for graph partitioning problems is not polynomial. The significance of the study is that this problem was solved using the software we have named KUSH, and a clustering method was proposed using the mathematical structure of spectral methods. The software KUSH introduced in this study prevents words from being treated as different simply because of the suffixes they receive, and thus can contribute to other text classification methods besides SSC. Our aim is for this software to be used in other work where texts are used as inputs. The study also presented the results of an empirical analysis demonstrating that graph partitioning methods can be used to solve text clustering problems, and the performance of the model proposed was evaluated using metrics commonly used in text clustering. The accuracy performance metric, which shows the overall effectiveness of clustering, indicated a 93.2% success level. The mean value of NMI, an entropy-based metric, was 67.52%. Another commonly used performance metric is F-score, also used in the present study. The mean F-score value was 93.32%. Finally, the mean value for the ARI performance metric was 74.73%. The study has also compared the performance of the SSC approach with the performance of the conventional k-means clustering algorithm. The SSC method was observed to produce better results. The promising results from the performance metrics, clearly visible in the relevant table, mean that the proposed method can be a useful and effective tool in text clustering. Based on the results of the empirical analysis, we believe that the software KUSH can be usefully employed prior to other categorization methods commonly used by researchers. Based on the method proposed by Fiedler, algebraic connectivity of the graphs representing the texts was utilized to categorize the texts. We were unable to find any studies in the text clustering literature reporting similarly high performance metrics. Moreover, we were unable to find any studies employing the concepts of the present study and making use of the TTC-3600 dataset, used in the present study for empirical analysis. The empirical analysis conducted with the TTC-3600 dataset produced promising and acceptable performance values. Spectral methods in general produce good results but cannot guarantee the ideal solution, which means that the SSC model proposed in the present study should be used together with the software tool KUSH. Otherwise, it is not possible to obtain the performance values reported in the relevant table and figure. These high-performance values are important in that they demonstrate how significant and successful the SSC method is when the software tool KUSH constitutes one of its stages..

DECLARATION OF ETHICAL STANDARDS

The authors of this article declare that the materials and methods used in this study do not require ethical committee permission and/or legal-special permission.

AUTHORS' CONTRIBUTIONS

Taner UÇKAN: Performed the experiments and analyse the results.

Cengiz HARK: Performed the experiments and analyse the results.

Ali KARCI: Performed the experiments and analyse the results.

CONFLICT OF INTEREST

There is no conflict of interest in this study.

REFERENCES

[1] Canbek G. and Ş. Sağıroğlu., "Bilgi ve Bilgisayar Güvenliği Casus Yazılımlarve Korunma Yöntemleri." ***Grafiker Yayınları***,Ankara, (2006).

[2] Durmaz, O. and H. Ş. Bilge., "Metin sınıflandırmada boyut azaltmanın etkileri ve özellik seçimi.", ***19th Conference on Signal Processing and Communications Applications (SIU 2011)***, Antalya, Turkey.(2011).

[3] Hark C.,Seyyarer E.,Uçkan T and Karı A., "Doğal dil İşleme yaklaşimlari ile yapisal olmayan dökümanlarin benzerliği.", ***2017 International Artificial Intelligence and Data Processing Symposium (IDAP)***. IEEE, (2017).

[4] Lewis DD., "Naive (Bayes) at forty: The independence assumption in information retrieval**.", *European conference on machine learning*.** **Springer**, Berlin, Heidelberg, (1998).

[5] Aas K. and Line E.,"Text categorisation: A survey." (1999).

[6] Andrews N.O. and Edward A.F., "Recent developments in document clustering", ***Department of Computer Science, Virginia Polytechnic Institute & State University***, (2007).

[7] Shah N. and Sunita M., "Document clustering: a detailed review.", ***International Journal of Applied Information Systems*** 4(5): 30-38, (2012).

[8] Pujari A.K., "*Data mining techniques",* **Universities press**, (2001).

[9] Bhushan S.N.B. and Ajit D., "Classification of text documents based on score level fusion approach." , ***Pattern Recognition Letters***, 94 :118-126, (2017).

[10] Wilson R.J. and John J.W.," Graphs: an introductory approach: a first course in discrete mathematics", ***John Wiley & Sons Inc***, (1990).

[11] Qiu H. and Edwin R.H., "Graph matching and clustering using spectral partitions." , ***Pattern Recognition*** ,39(1): 22-34, (2006).

[12] Kılınç D., Özçift A., Bozyigit F., Yıldırım P., Yücalar, F. And Borandag E.," TTC-3600: A new benchmark dataset for Turkish text categorization",***Journal of Information Science***, 43(2): 174–185, (2017).

[13] Uçkan T. and Karcı A., "Extractive multi-document text summarization based on graph independent sets.", ***Egyptian Informatics Journal,*** (2020).

[14] Hark C., Seyyarer, A., Uçkan T. And Karci A., "Doğal dil İşleme yaklaşimlari ile yapisal olmayan dökümanlarin benzerliği.", ***2017 International Artificial Intelligence and Data Processing Symposium (IDAP)***. IEEE, (2017).

[15] Hark C., Uçkan T., Seyyarer A. and Karci A.,"Metin Özetleme İçin Çizge Tabanlı Bir Öneri.", ***IDAP 2018– international artificial intelligence and data processing symposium***, (2018).

[16]Karcı A., "Çizge algoritmaları ve çizge bölmeleme/Graph algorithms and graph partitioning.", (1998).

[17] Von L.U., "A tutorial on spectral clustering.", ***Statistics and computing*** ,17(4): 395-416, (2007).

[18] Slininger B., "Fiedlers theory of spectral graph partitioning.", ***Dosegljivo: http://citeseerx. ist. psu. edu/viewdoc/download***, (2013).

[19] Jia H., Shifei D. and Mingjing D., "Self-tuning p-spectral clustering based on shared nearest neighbors." , ***Cognitive computation*** ,7(5): 622-632, (2015).

[20] MacDonald J. K. L., "Successive approximations by the Rayleigh-Ritz variation method.", ***Physical Review*** ,43.10 :830, (1933).

[21] Hagen L. and Andrew B.K., "New spectral methods for ratio cut partitioning and clustering.", ***IEEE transactions on computer-aided design of integrated circuits and systems*** ,11(9):1074-1085, (1992).

[22] Shi J. and Jitendra M., "Normalized cuts and image segmentation.", ***IEEE Transactions on pattern analysis and machine intelligence***, 22(8): 888-905, (2000).

[23] Ayed A.B., Mohamed B.H. and Adel M. Alimi., "Adaptive fuzzy exponent cluster ensemble system based feature selection and spectral clustering." , ***2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)***, (2017).

[24] Filippone M., Camastra F., Masulli F. and Rovetta S., "A survey of kernel and spectral methods for clustering.", ***Pattern recognition*** ,41(1): 176-190, (2008).

[25] Casper W. R. and Balu N., "A new spectral clustering algorithm"***, arXiv preprint*** *arXiv:1710.02756* (2017).

[26] Alguliev R.M., Ramiz M.A. and Makrufa S.H., "GenDocSum+ MCLR: Generic document summarization based on maximum coverage and less redundancy." , ***Expert Systems with Applications***, 39(16): 12460-12473, (2012).

[27] Fiedler M., "Algebraic connectivity of graphs, Czechoslovakian Math.", 298-305, (1973).

[28] Chung F.RK., "Lectures on spectral graph theory"***, CBMS Lectures,*** 6(92): 17-21, (1996).

[29] Zheng C.T., Cheng L. and Hau S.W., "Corpus-based topic diffusion for short text clustering.", ***Neurocomputing*** ,275: 2444-2458, (2018).

[30] Xu J., Xu B., Wang P., Zheng S., Tian G. and Zhao, J."Self-taught convolutional neural networks for short text clustering.", ***Neural Networks,*** 88: 22-31, (2017).

[31] Beil F., Martin E. and Xiaowei X., "Frequent term-based text clustering.", ***Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining***, (2002).

[32] Schütze H., Christopher D.M. and Prabhakar R.,"Introduction to information retrieval", ***Cambridge: Cambridge University Press***, (2008).

[33] Kozlowski M. and Henryk R., "Clustering of semantically enriched short texts.", ***Journal of Intelligent Information Systems,*** 53(1): 69-92, (2019).