Hybrid Harmony Search Algorithm to Solve the Feature Selection for Data Mining Applications

Chapter · July 2020

DOI: 10.1002/9781119551621.ch2

2

Hybrid Harmony Search Algorithm to Solve the Feature Selection for Data Mining Applications

Laith Mohammad Abualigah[1], *Mofleh Al-diabat*[2], *Mohammad Al Shinwan*[3], *Khaldoon Dhou*[4], *Bisan Alsalibi*[5], *Essam Said Hanandeh*[6], *and Mohammad Shehab*[7]

[1] *Faculty of Computer Sciences and Informatics, Amman Arab University, Jordan*
[2] *Department of Computer Science, Al Albayt University, Jordan*
[3] *Faculty of Computer Sciences and Informatics, Amman Arab University, Jordan*
[4] *Breech School of Business, Drury University, Springfield, MO, United States*
[5] *School of Computer Sciences, Universiti Sains Malaysia, Malaysia*
[6] *Department of Computer Information System, Zarqa University, Jordan*
[7] *Computer Science Department, Aqaba University of Technology, Aqaba, Jordan*

2.1 Introduction

Lately, the increasing size of all sorts text and data information on websites makes the method of text clustering (TC) a lot more complicated. The TC technique is employed to cluster an enormous variety of documents into a set of intelligible and connected clusters [1, 2]. Usually, TC is employed in several domains like text mining, data processing, pattern recognition, image clustering [3, 4].

The vector house model (VSM) may be a mutual arrangement pattern utilized in the text field to simplify a document's parts as an array [i.e., row of features (vectors)]. Consequently, the entire document is painted as a vector of features or terms score (weighing values), and every term weight price is painted within a row (dimension space). For this reason, a multi-dimensional house may have mutual and massive downside issues over the text clustering technique. This downside affects the TC method by reducing its performance and increasing the general system runtime [5].

Usually, any text documents containing informative and uninformative features, wherever associate degree uninformative is as extraneous, redundant, and uniformly distributed feature [6, 7]. These varieties of features cut back on the execution of the application of the clustering technique and affect the method in a very dangerous manner because every document contains several extraneous features. An unsupervised feature section (FS) may be a common downside in this domain, and it's a crucial assignment wont to realize a replacement set of feature to reinforce the accuracy of the text clustering rule. This method is enforced with no premonition of the document's category label. A feature choice downside is outlined as associate degree optimization downside with two constraints (i.e., two objective measures: minimizing the involved text feature and maximizing the performance

Recent Advances in Hybrid Metaheuristics for Data Clustering, First Edition.
Edited by Sourav De, Sandip Dey, and Siddhartha Bhattacharyya.

value of the implied clustering algorithm) [5]. Numerous scopes within the text mining field profit from the feature choice technique like the image analysis and clustering applications [8], text clustering [9], cancer classification [10], world numerical optimization issues [11], systems management [7], image cryptography and lossless compression [12], cistron choice [13], and data retrieval [14].

Text pages have become a necessary resource within electronic sites that contain an unorganized large quantity of text documents; these sites include news sites, minutes, reports, and science info digital libraries [15]. TC is a lively information approach that partitions several documents into some clusters, each containing similar text. This method makes sites easier to access, clearer to understand, and more organized. Any TC algorithmic program tries to seek out coherent clusters by partitioning the text documents into clusters so as to assign every document to the best cluster supported by the document contents [16].

Harmony search (HS) is a kind of robust meta-heuristic algorithm propositioned by Prof. Zong Woo Geem in 2001 [17]. It follows the music discovery method and has had success tackling several laborious improvement issues such as vanishing purpose detection for self-driving automobiles [18], numerical performance improvement [19], text document clustering [2], optimum power flow [20], and timetabling issues [21, 22].

Harmony search is one of the stronger algorithms within the domain that can resolve several random issues, particularly for acquiring quality subsets of features for reconnoitering the entire benchmark datasets. A completely unique feature choice exploitation is used by the harmony search technique for extracting a replacement set of informative features, and it had been used with several feature subsets [23]. Harmony search integrity was utilized to scale back the runtime quality and enhance the accuracy of the feature choice in terms of the method performance. Experimental results exhibited that the planned feature choice technique improved the performance of the text cluster method.

The genetic algorithmic program (GA) is will settle on a singular set of informative features for developing the execution of the text clustering process. This technique uses the frequency-inverse document frequency (TF–IDF) as a coefficient score or theme to cut back every term relationship [24]. Experiment analysis was applied on text spam email. The results said that the planned genetic algorithmic program for finding the feature choice drawback improved the performance value of the clustering process.

The downside of feature choice may be an optimization problem that is employed to get a replacement set of excellent features [23]. The cat swarm optimization (CSO) formula has been introduced additionally to reinforce optimization issues. Yet, Cat swarm optimization is restricted to long runtimes. They modify the formula to extend and improve the feature choice downside within the method of the text classification [25]. Experiment results showed that the planned changed formula (cat swarm optimization) succeeds and got better results in comparison with the initial version (cat swarm optimization) and got a lot of correct results in feature choice method than victimization TF-IDF alone.

One of the distinctive feature choice techniques has been projected by exploitation the harmony search rule for selecting and obtaining a replacement set of informative knowledge feature [23]. The projected rule during this paper was to cut back the runtime of the system and to decrease the uninformative knowledge feature. Finally, their results said that the projected modification of the harmony search rule enriched the performance value of the feature choice method in regard to the correct set, owing to its fine features.

Particle swarm improvement (PSO) may be a powerful formula projected by Kennedy, Eberhart. It selects a singular set of a lot of informative features for developing the performance of the text clustering. This technique uses the term frequency-inverse document frequency (TF–IDF) as a weight theme for cutting back every term relationship [26]. Experiment analysis and discussion were made in huge Brobdingnagian text documents. The results exhibited that the projected particle swarm improvement methodology for finding the text feature choice downside got better results.

Recently, several optimization algorithms are applied with success to resolve varied straightforward and onerous optimization issues [9, 27–31]. In this paper, we tend to introduced a unique feature choice technique exploitation the harmony search algorithmic rule, namely, FSHSTC. This approach is applied to choose a definite set of fine informative features for making the text agglomeration technique making productive. The biggest objective of this paper is to replace the term or feature choice technique for supporting the performance value of the text agglomeration procedure. Experimental results and its analysis were conducted on four varied datasets to prove and check the utilized algorithmic rule. The results showed that the FSHSTC got better results in comparison with the opposite comparative ways in terms of two evaluation measures (i.e., F-measures and accuracy).

The structure of this paper is provided as follows: Section 2 describes the overall analysis framework for explaining the feature choice downside. Section 3 shows a way to prepare the text clustering victimization of the pre-processing steps. Section 4 presents, however, that the feature choice downside enforces victimizaton on the essential harmony search rule. Section 5 shows the essential steps of the projected hybrid harmony search rule for feature choice downside. Section 6 defines the best suggestions the text clustering technique. Section 7 illustrates the best suggestions for the k-means clustering rule. Experiments analysis and the results are provided in Section 8. Finally, Section 9 gives our conclusions.

2.2 Research Framework

The text preprocessing steps have been adjusted to choose an optimum solution by giving a replacement set of options (informative features) for increasing the performance of the underlying text document agglomeration algorithms and cutting back on its procedure time. Consequently, this paper projected associate economical and effective classification to pronounce a replacement optimum set of text options so as to comprehend the paper's aims. In the end, these techniques are enforced so as to alter the rule of text agglomeration in keeping with the order of many steps. Figure 2.1 shows the methodology of the projected hybrid feature section technique supported by the harmony search rule for the text agglomeration method (H-HSA).

In the beginning, the preprocessing steps are utilized to prepare the document within the kind of numerical style (data). In the second step, the harmony search rule is adjusted to resolve the feature choice drawback by reducing uninformative options and come up with a replacement set of fine options. The feature choice procedure is examined as a preprocessing step in pattern analysis and recognition, computing, machine unsupervised learning, etc. It is applied as a decision-maker to decide which set of informative text features is most

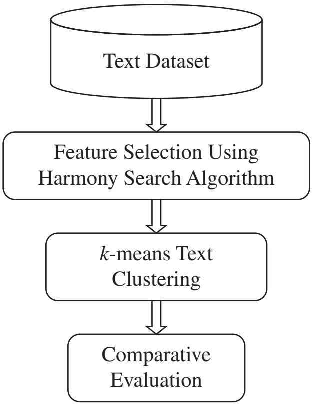
Text Dataset
Feature Selection Using Harmony Search Algorithm
k-means Text Clustering
Comparative Evaluation

Figure 2.1 Research framework of the proposed hybrid method

efficient by excluding uninformative text features. Then, within the third step, the *k*-means text agglomeration technique is used to assign every document to the fitting cluster. The *k*-means text agglomeration rule is best as a result of it being a wise text analysis technique to assess the performance of the feature choice methodology to improve the projected harmony search rule.

2.3 Text Preprocessing

TC needs text design and representation [32]. The linguistic communication process (NLP) is a technology used in interactions linking humans and PC languages. This method is critical and could be a critical step in the text mining domain. It examines the fundamental text preprocessing to acquire document options by processing a variety of symbols like words, thereby removing stop words for a text illustration [2]. The projected technique relies on the text FS domain and TD cluster classifications domain, just like the preprocessing for the illustration of TD. The preprocessing levels are divided into four major levels or processes. The first one is tokenization, the second one is stop word removal, the third one is stemming, and finally the last one is text representation [5, 33].

2.3.1 Tokenization

Tokenization is the task of isolating words into tokens, likely losing a few assortments, like accentuation. These tokens are in some cases connected to terms or words; in any case, it's fundamental to make a type/token refinement. A token is a word illustration of an arrangement of characters in an archive that's joined to a supportive syntactic unit. A sort means the assortment of all tokens alongside the indistinguishable character arrangement. A term may be a sort that's consolidated inside the data recovery system's lexicon [34].

2.3.2 Stop Words Removal

Stop words are current common words, like *in*, *no*, *an*, *that*, *yes* and *a few*, encouraged as elective common words that are part of ordinarily utilized and minor accommodating words inside the TD. These words ought to be far away from content archives as they ordinarily have tall recurrence, which decreases the execution of the TC strategy. The list[1] of stop words contains a total of 571 words [5].

2.3.3 Stemming

Stemming is the strategy of diminishing curved terms to their term stem (root). The stem running the show isn't indistinguishable from the morphological root handle; it outlines words to the indistinguishable stem, whether or not this stem isn't in itself an immaculate root. The Porter (http://tartarus.org/martin/PorterStemmer/) stemmer is the common stemming method embraced in content mining [2, 34]. All content preprocessing steps come from Python NLTK demos for tongue content processing.[2]

2.3.4 Text Document Representation

Vector space model (VSM) is a good example of TDs in an ordinary use [15]. It comes from the early seventies. Each archive is spoken to as a vector of term weight to encourage character calculation. Each term inside the set speaks to a measurement of the weighted worth to create the standard of the content elucidation algorithm running the show and to scale back the time estimation.

The term coefficient is used by the vector show model (VSM) to point out the TDs in an exceedingly ordinary organization, as in Eq. (2.1). This show speaks to each archive as a vector, as in Eq. (2.2) [16]. Eq. (2.1) speaks to n reports and t terms, as in:

$$VSM = \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,(t-1)} & w_{1,t} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ w_{(n-1),1} & w_{(n-1),2} & \cdots & \cdots & w_{(n-1),t} \\ w_{n,1} & w_{n,2} & \cdots & w_{n,(t-1)} & w_{n,t} \end{bmatrix} \tag{2.1}$$

$$d_i = (w_{i,1}, w_{i,2}, ..., w_{i,j}, ..., w_{i,t}), \tag{2.2}$$

2.3.5 Term Weight (TF-IDF)

Term weight is a critical numerical datum used to consider the weight of record words (highlights or terms) for TD agglomeration forms in line with the term reiteration [35]. A common term weighting subject in text mining is term frequency-inverse archive recurrence (TF-IDF) [3, 36].

TF-IDF may well be a conventional weight topic that's similar to gage term weight in content mining for archive outlines. Each archive is portrayed as a vector of term weight [5]. The term weight is calculated by Eq. (2.3), as shown here:

$$w_{i,j} = tf(i,j) * log(n/df(j)), \tag{2.3}$$

[1] http://www.unine.ch/Info/clef/

[2] http://text-processing.com/demo/

where emph$w_{i,j}$ speaks to the weight of term *j* within the report number *i*, *tf(i,j)* is the occurrence of term *j* within the record number *i*, *n* is the number of all reports in the dataset, and *df(j)* is the number of records that contain the term number *j* [36].

2.4 Text Feature Selection

This area clarifies the proposed harmony that is used in the calculation of tackling the FS problem.

2.4.1 Mathematical Model of the Feature Selection Problem

The feature selection issue can defined as an NP-Hard issue in the form of an optimization issue to create an ideal subset of more enlightened highlights. Given f_i a collection of content highlights $f_i = f_{i1}, f_{i2}, \ldots, f_{it}$, where *t* is the number of all interesting content highlights in all archives, and *i* is the number of a report. Let sf_i be an unused subset of instructive content features $sf_i = s_{i1}, s_{i2}, \ldots, s_{ij}, s_{im}$, *m* may be a unused special number of all highlights in all reports, in the event that $s_{ij} = 1$ suggests the j_{th} highlight is chosen as an enlightening include in archive *i*, in the event that $s_{ij} = 0$ suggests the j_{th} include is an informative in report *i* [2, 5, 37–39].

2.4.2 Solution Representation

In this paper, the include determination strategy is based on the agreement look calculation, which begins with irregular arrangements (arbitrary starting arrangements) and progresses with the populace (arrangements) by getting a universally ideal arrangement [5, 7, 39]. Each special highlight (one position) within the archive considers the measurement of the look space. Table 2.1 presents the arrangement of the include determination issue [40].

Note, *X* speaks to a single arrangement of the included determination issue. In the event that the esteem of position number *j* (j_{th}) is *1*, the j_{th} highlight is chosen in this arrangement as an instructive highlight. On the off-chance that the esteem of position number *j* (j_{th}) is *0*, the j_{th} include isn't chosen in this arrangements as an instructive include (uninformative/non-informative), at long last, on the off-chance that the esteem of position number *j* (j_{th}) −1, the j_{th} include was not included within the unique content archive at the start.

2.4.3 Fitness Function

The fitness function (FF) is a particular type of objective function that summarizes, as shown in Table 2.1, how close a given design solution is to achieving the set aims [41]. Inside

Table 2.1 Feature selection solution representation

X	*0*	*1*	*1*	*−1*	*−1*	*1*	*0*	*−1*	*1*	*−1*

the concordance look algorithmic program for finding the highlight choice drawback, the cruel supreme qualification (frantic) is utilized as a fitness perform. This fitness perform is predicated on using the coefficient subject (TF-IDF) as the relate degree objective perform for each position. Cruel outright refinement may be a common way to find the highlight choice drawback in this space to allot a significance score for alternatives by plotting the qualification between the standard steady ones and the alternatives weight scores [34]. At that point, computing the refinement between the cruel and middle of $x_{i,j}$ [36, 42] looks like the following equation:

$$MAD_{(Xi)} = \frac{1}{a_i} \sum_{j=1}^{t} |x_{i,j} - \bar{x}_i|, \tag{2.4}$$

where,

$$\bar{x}_i = \left(\frac{1}{a}\right) \sum_{j=1}^{t} x_{ij}, \tag{2.5}$$

$x_{i,j}$ is the current value of the feature in position number j in document number i, a_i is the number of selected features in document number i, t is the number of all unique features in the dataset, and $\bar{x}_i$ is the mean of vector i (document number i).

2.5 Harmony Search Algorithm

The harmony search rule produces (generates) irregular harmony memory (HM), which contains a collection of candidate solutions. The harmony search rule then reinforces the harmony memory to attain the best answer to unravel the matter (associate degree best set of a lot of informative features). Every position or musician (unique term) may be a dimension within the search area. The solutions of the harmony search that are assessed by the mentioned fitness function operate as in the equivalent weight of Eq. 2.4, it's accustomed get associate degree best harmony (global best solution) [7, 17]. The harmony search rule provides the solutions in five main steps. covered next.

2.5.1 Parameters Initialization

The feature choice drawback is delineated as associate optimization drawback by means of attempting to find the maximize worth of the fitness operatation $f_{(x)}$, where x_i is the i_{th} position worth. Some parameters are employed in the harmony search algorithmic rule for feature choice drawback supported literature studies as harmony memory solutions=50 (number of solutions), concordance memory thought rate (HMCR)=0.9 is that the probability of choosing the choice variable whether or not from memory or each which way, least pitch altering rate (PARmin)=0.45 is that the least pitch altering rate, most pitch altering rate (PARmax)=0.9 is that the foremost pitch altering rate, bwmin=0.1 is that the least data measure, bwmax=1 is that the foremost data degree, and (NI) is that the assortment of all cycles [7, 21, 43].

2.5.2 Harmony Memory Initialization

The calculation incorporates a store of arrangements within the harmony memory (HM) lattice that's filled by creating HMS haphazardly, as shown here:

$$\mathbf{HMS} = \begin{bmatrix} x_1^1 & \cdots & x_{t-1}^1 & x_t^1 \\ x_1^2 & \cdots & x_{t-1}^2 & x_t^2 \\ \vdots & \ddots & \vdots & \vdots \\ x_1^{HMS-1} & \cdots & x_{t-1}^{HMS-1} & x_t^{HMS-1} \\ x_1^{HMS} & \cdots & x_{t-1}^{HMS} & x_t^{HMS} \end{bmatrix} \tag{2.6}$$

$$x_i^j = LB_i + Rand * (UB_i - LB_i), \qquad j = 1, 2, ..., HMS. \tag{2.7}$$

The agreement harmony memory is made as a lattice in Eqn. 2.6, where $[LB_i]$ is the lower bound and $[UB_i\ ]$ is the upper bound.

2.5.3 Generating a New Solution

Generating a new solution follows these three rules: memory consideration, pitch adjustment, and random selection in Algorithm 1 [7, 16].

$$X' = (x_1', x_2', ..., x_t') \tag{2.8}$$

Arrangements of the concordance look calculation are powerfully merged, agreeing to the $PAR(I)$ esteem and $bw(I)$ values [7, 16, 22]. If an arbitrary number between [0, 1] is created and decreased or increased to the likelihood of standard, then the modern choice variable for the following iteration (x) is decided $'$:

$$PAR(I) = PAR_{min} + \left(\frac{PAR_{max} - PAR_{min}}{I_{max}} \right) * I, \tag{2.9}$$

where

$$bw(I) = bw_{max} exp \left(\frac{In \left(\frac{bw_{min}}{bw_{max}} \right)}{I_{max}} \right) * I, \tag{2.10}$$

Algorithm 1 Improvise a new solution

```
Input: Harmony memory $HM$ solutions
Output: A new solution as vector represented in 2.8
for each $j \in [1, t]$ do
    if Rand (0,1) $\leq$ HMCR then
        $x_j' = HM[i][j] wherei \sim U(1, 2, ...., HMS)$
        if Rand(0,1) $\leq$ PAR then
            $x_j' = x_j' \pm rand\times$ bw, where r~ U(0,1) and bw is distance bandwidth

            else $x_i' = LBj + rand \times (UBj - LBj)$
        end if
    end if
end for
```

$PAR(I)$ is the pitch altering rate for a modern arrangement, I is the number of the current cycle, I_{max} is the max number of cycles, bw_{min} is the least altering rate, and bw_{max} is the most extreme altering rate.

2.5.4 Update Harmony Memory

In the event that the modern arrangement has way better wellness or fitness, the work esteem will replace the most noticeably awful concordance arrangement.

2.5.5 Check the Stopping Criterion

When the agreement look calculation comes to the greatest number of iterations, it will halt, and the *HMCR* and *PAR* parameters of the concordance look that offer assistance to seek for the universal and local arrangements [7, 16, 19].

2.6 Text Clustering

In this section, we show the steps of the content cluster method after making a fresh set of choices by changing the concordance look algorithmic program to improve the content cluster procedure (FSHSTC), improve the *k*-mean algorithmic program, and update the cluster centers of mass and similarity.

2.6.1 Mathematical Model of the Text Clustering

The content clustering method is portrayed as follows: given D and a bunch of content archives $D = d_1, d_2, ..., d_j, ..., d_n$, D is the number of all records inside the dataset (reports), and d_1 denotes the favorite document, Cos_{di} is an objective work that is an irrelevant record based on d_i. By and large, these common features judge the execution of the content clustering strategy [16, 44].

2.6.2 Find Clusters Centroid

In arrange to segment (cluster or gather) a set of colossal records into a subset of significant clusters, within each cycle, the cluster centroids are overhauled according to the substance of the clusters. Each archive within the dataset is compared to the comparable cluster centroid (to one cluster). Here, ck is the cluster centroid $ck = (ck1, ck2, ..., ckj, ..., ckK)$, and ckj is the centroid of cluster number j [16, 45]. This is what it looks like after modifying the cluster's centroid:

$$c_{kj} = \frac{\sum_{i=1}^{n}(a_{ki})d_i}{\sum_{j=1}^{r_i} a_{kj}}, \tag{2.11}$$

where d_i is the report i that has a place to the c_j centroid of the cluster j, a_{kj} is the number of reports that have a place to cluster j, and r_i is the number of reports in cluster i [16].

2.6.3 Similarity Measure

Cosine closeness is the common closeness degree utilized within the content report clustering procedure to calculate the likeness between two vectors; here, d_1 is record number *1*, and d_2 is the cluster centroid. This is the equation:

$$Cos(d_1, d_2) = \frac{\sum_{j=1}^{t} w(t_j, d_1) \times w(t_j, d_2)}{\sqrt{\sum_{j=1}^{t} w(t_j, d_1)^2}\sqrt{\sum_{j=1}^{t} w(t_j, d_2)^2}}, \tag{2.12}$$

where $w_{tj,d1}$ is the weight score of term number *j* within the archive number 1, $w_{tj,d2}$ is the weight score of term number *j* within the archive number 2 $\sum_{j=1}^{t} w(t_j, d_1)^2$ is the summation of all terms' weight within record 1 under square from term number $\{j = 1 \text{ to } t\}$, and $\sum_{j=1}^{t} w(t_j, d_2)^2$ is the summation of all terms' weight within record 2 under square from term number $\{j = 1 \text{ to } t\}$, and d_2 speak is the cluster centroid [42, 45].

2.7 *k*-means text clustering algorithm

The *k*-means clump was introduced in 1967 as an area search clump rule [46]. It is a clustering rule utilized in the space of the text document clump; as a result, it is thought of as the correct clustering rule to settle on the initial cluster centroids in this area. The *k*-means is utilized to parcel (cluster) a set of content reports with a multidimensional region and bounty of region content highlights $D = (d_1, d_2, d_3, ..., d_n)$ into comparable and associated clusters. The *k*-means content clump runs the show using the most extreme likeness as a likeness content for tasks in each record of the comparative cluster by proportionate weight; see Eq. 2.12. It uses X as the information lattice $n * k$, where n is the number of all reports within the given dataset, K is the number of all clusters as predefined, k is the cluster centroid number, each report within the dataset is a vector of term weight scores $d_i = (w_{i1}, w_{i2}, w_{i3}, ..., w_{it})$, t is the number of all interesting highlights within the dataset D, *k*-means calculation looks approximately like the ideal $n * k$ [16, 35].

The *k*-means clustering strategy takes place in these five primary steps:

1) Each record inside the dataset is a vector of terms weight score.
2) Initialize the centroids of the clusters randomly with the reports matrix X.
3) Work out the likeness score or worth for each record with the middle of the clusters dividing each archive into a parcel of associated centroids by the circular work closeness in Eq. 2.11.
4) Update the center of the clusters with the updated documents that relate to every cluster center of mass in line with the current state of affairs for obtaining the correct cluster centroid exploitation of two weighting schemes [47].

Algorithm 2 k-means clustering algorithm

```
Input: $D$ is huge collection of documents, $K$ is the number of all clusters.
Output: Assign $D$ to $K$.
 Termination criteria
Randomly choosing $K$ document as clusters centroid $C = (c_1, c_2, ...., c_K)$
Initialize matrix X as zeros
for all $d$ in $D$ do
    let $j$= $arg_{min}$ $k$ based on $Cos(d_i, c_k)$
end for
Update the centroids of the clusters using Eqn. 2.11
End
```

2.8 Experimental Results

We have programmed the entire system (i.e., harmony search algorithmic rule for determination the feature choice downside and k-means for text clump problem) using the Matlab software package (version seven 7.10.0). We will outline each dataset's elements and organization, make a case for the examination criteria, and show the results.

Table 2.2 presents the datasets utilized within the test of this chapter. There are seven typical benchmark datasets we will explore and compare the execution of the k-means algorithmic for the include choice strategy and the include choice procedure. The datasets are available at (http://sites.labic.icmc.usp.br/text_collections). The main dataset (DS1), known as CSTR, contains 100 irregular archives from the abstracts of specialized reports that has a place to 2 points. The second dataset (DS2), known as Twenty Newsgroups, contains 200 arbitrary records from completely distinctive newsgroups that have a place to four themes. The third dataset (DS3), known as Domz-business, contains 250 arbitrary archives from a web catalog of web assets that has a place to 6 subjects. The fourth dataset (DS4), known as Domz-computer, contains 300 arbitrary records from completely distinctive newsgroups that have a place to 10 themes. The fifth dataset (DS5), known as Reuters–21578, contains 350 arbitrary archives from the newswire from 1987 that have a place to 10 subjects. The 6th dataset (DS6), called ACM, contains 200 arbitrary archives from an affiliation for computing apparatus that has a place to 10 points. The seventh dataset (DS7), known as WebAce, contain 1,560 irregular archives from a web specialist for record categorization and investigation that has a place to 15 points.

2.8.1 Evaluation Measures

The comparative assessments were done utilizing one inner assessment measure, likeness degree, and two outside assessment measures, precision (Ac) and F-measure (F). These

Table 2.2 Text Datasets Characteristics

Datasets	Source	Number of Documents	Number of Terms	Number of Clusters
DS1	CSTR	100	1260	2
With FS			533	
DS2	20Newsgroup	200	6518	4
With FS			1051	
DS3	Domz-Businss	250	1156	6
With FS			111	
DS4	Domz-Computer	300	1247	8
With FS			214	
DS5	Reuters-21578	350	3258	10
With FS			1048	
DS6	ACM	350	24457	10
With FS			9891	
DS7	WebAce	1560	8880	20
With FS			4540	

measures are common assessment criteria utilized in the content clustering space to assess the clusters precisely [36].

2.8.1.1 F-measure Based on Clustering Evaluation

The F-measure (F) could be a common judgment utilized within the content clustering region. It controls the degree of truth clusters and depends on two judgments: exactness (P) and review (R) [47]. The F-measure controls for the cluster j and lesson (document) i is decided by the following:

$$P(i,j) = \frac{n_{i,j}}{n_j}, \quad R(i,j) = \frac{n_{i,j}}{n_i}, \tag{2.13}$$

where n_{ij} is the number of individuals of course i in cluster j, n_j is the number of individuals of cluster j, and n_i is the number of individuals of lesson i.

$$F(i,j) = \frac{2P(i,j)R(i,j)}{P(i,j)R(i,j)}, \tag{2.14}$$

where *P(i,j)* is the accuracy of individuals of course i in cluster j, *R(i,j)* is the review of individuals of course i in cluster j, and the F-measure for all is clusters is calculated by the following:

$$F = \sum_j \frac{n_j}{n} \max_i \{n(i,j)\}, \tag{2.15}$$

2.8.1.2 Accuracy Based on Clustering Evaluation

Precision (AC) is the common outside estimation that is utilized accurately to find out the rate of the archives of each cluster related to the following [47]:

$$Ac = \sum_{i=1}^{k} \frac{1}{n} P(i,j) \tag{2.16}$$

where $P(i,j)$ is the exactness esteem for lesson i in cluster j, n is the number of all records in each cluster, and k is the number of all clusters.

2.8.2 Results and Discussions

This paper utilized the k-means algorithmic rule to resolve the text agglomeration downside supported by two methods. The first one was the k-means text agglomeration using the feature choice technique, namely, KMTC. In the second method, k-means is applied to resolve text agglomeration by using the feature choice ways exploitation harmony search algorithmic rule, namely, FSHSTC.

The anticipated k-means content agglomeration with the highlight choice method was better than the k-means content agglomeration strategy. It did not include the choice procedure inside DS1, DS3, DS5, DS6, and DS7 in general estimations, but it made strides with the F-measure in DS2 and DS4. We tried to misuse seven common datasets that appeared in Table 2.1. Note that applying the feature choice technique before the text agglomeration technique was beneficial to enhance the agglomeration method by handling an occasional variety of options, which makes the agglomeration procedure easier to use to partition an enormous number of text documents.

For accurate results, we performed the experiments more than 20 times, and the harmony search is an international search algorithmic rule that runs 500 iterations in every run. This extensiveness is supported by the literature that validates the projected technique (FSHSTC). When using 500 cycles, the world look algorithmic had better results for the include choice strategy. The k-means may be a better local look algorithmic. So, 100 cycles is enough for the local look algorithmic to run the show for the content archive agglomeration procedure [16].

Table 2.3 shows that the execution of different methods depends on a cluster's quality. For the most part, the arranged FS method doesn't perform well with the look equation, and but content clustering method upheld the estimations that utilized over the seven benchmark content datasets. The choice of dataset is a vital step still for the content clustering strategy to get fewer content alternatives for the best record clusters. Rationally, any system that deals with the related features will perform the text analysis with efficiency.

The arranged FSHSTC performd okay after content clustering, and it diminished the number of choices. In other words, it scaled back the number of alternatives in DS1 (1,260 to 533), scaled back the number of choices in DS2 (6,518 to 1,051), scaled back the number of choices in DS3 (1156 to 111), scaled back the number of alternatives in DS4 (1,247 to 214), scaled back the number of alternatives in DS5 (3258 to 1,048), scaled back the number of

Table 2.3 The Algorithm Efficacy Based on Clusters' Quality Results

Dataset	Method	KMTC	FSHSTC
DS1			
	Accuracy	0.5800	**0.6060**
	F-measure	0.5795	**0.5808**
	Rank	2	1
DS2			
	Accuracy	0.3630	**0.3755**
	F-measure	**0.3782**	0.3774
	Partial rank	1	1
DS3			
	Accuracy	0.3578	**0.3687**
	F-measure	0.3478	**0.3625**
	Partial rank	2	1
DS4			
	Accuracy	0.2714	**0.2854**
	F-measure	**0.2584**	0.2558
	Partial rank	1	1
DS5			
	Accuracy	0.4509	**0.4688**
	F-measure	0.3855	**0.4005**
	Partial rank	2	1
DS6			
	Accuracy	0.3857	**0.3989**
	F-measure	0.3140	**0.3362**
	Partial rank	2	1
DS7			
	Accuracy	0.4822	**0.4954**
	F-measure	0.4350	**0.4440**
	Final ranking	2	1
Mean rank		1.71	**1**
Final rank		2	**1**

choices in DS6 (24,457 to 9,891), and scaled back the number of alternatives in DS7 (8,880 to 4,540). The diminishing assortment of alternatives is influenced by the execution time to be less and increment the fundamental clustering execution. The arranged FSHSTC overcomes the *k*-means stand-alone in all the cases.

Figure 2.2 and Figure 2.3 shows that the execution (precision and F-measure) of the anticipated method (FSHSTC) upheld its clusters' quality using seven standard content

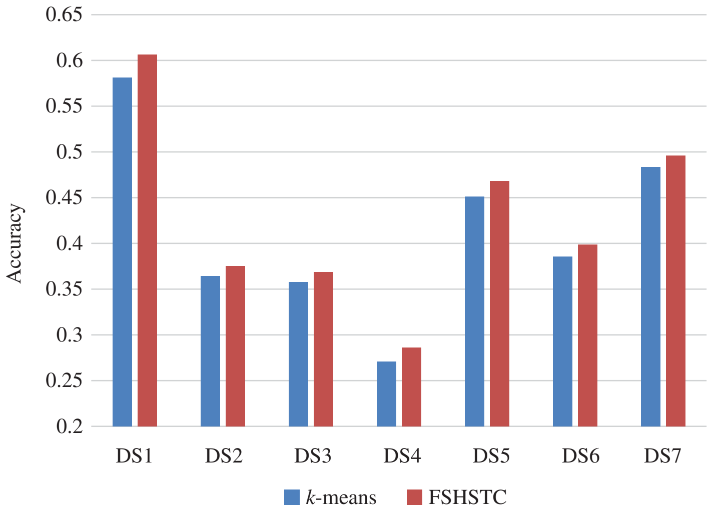

Figure 2.2 The accuracy of the *k*-means text clustering methods

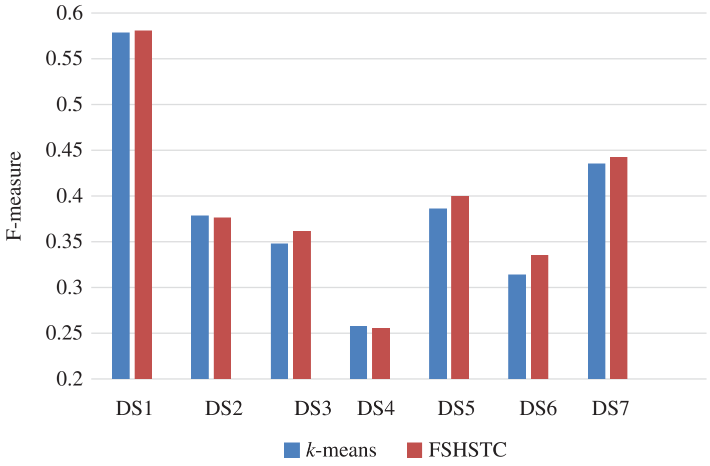

Figure 2.3 The F-measure score of the *k*-means technique

datasets. It's clear that the anticipated FSHSTC performed affirmatively and overcame the inverse comparative procedure (*k*-means clump stand-alone) bolstered by the content clump method with the clump exactness and F-measure.

The projected FSHSTC scored better on accuracy and F- as an external measure in all the datasets. All experiments produced better results once the projected system applied FSHSTC to support the two measurements expected for DS2 and DS4; however, the results virtually are identical. Also, there were no improvements even with a shorter execution time (runtime). Finally, the anticipated method (i.e., FSHSTC) was better than the inverse associated procedure and obtains the results.

2.9 Conclusion

In this chapter, the modern highlight choice strategy is utilized with the concordance look algorithmic program to look for an ideal unused subset of data to make the clustering strategy successfully by getting extra adjusted clusters. This technique was presented so as to improve the execution of the content clustering procedure. This unused strategy is called the highlight choice strategy misuse concordance look algorithmic program for the content were clustering procedure (FSHSTC), which overcomes the *k*-means clustering algorithmic program by improving the execution of the content clustering algorithmic program. FSHSTC was surveyed using numerous benchmark content datasets (seven benchmark datasets). The results were that the execution content clustering may be developed and make strides with the anticipated highlight choice technique. For future work, the anticipated FSHSTC may be changed as a greenhorn adaptation (adjusted, crossbreed, and improved) to upgrade the global investigation of the content clustering technique.

References

1 Abualigah, L.M., Khader, A.T., and Hanandeh, E.S. (2016) A combination of objective functions and hybrid krill herd algorithm for text document clustering analysis. *Engineering Applications of Artificial Intelligence*.

2 Bharti, K.K. and Singh, P.K. (2015) Hybrid dimension reduction by integrating feature selection with feature extraction method for text clustering. *Expert Systems with Applications*, **42** (6), 3105–3114.

3 Wang, X., Cao, J., Liu, Y., Gao, S., and Deng, X. (2012) Text clustering based on the improved tf-idf by the iterative algorithm, in *Electrical & Electronics Engineering (EEESYM), 2012 IEEE Symposium on*, IEEE, pp. 140–143.

4 Mouring, M., Dhou, K., and Hadzikadic, M. (2018) A novel algorithm for bi-level image coding and lossless compression based on virtual ant colonies., in *COMPLEXIS*, pp. 72–78.

5 Bharti, K.K. and Singh, P.K. (2016) Opposition chaotic fitness mutation based adaptive inertia weight bpso for feature selection in text clustering. *Applied Soft Computing*, **43**, 20–34.

6 Al-Sai, Z.A. and Abualigah, L.M. (2017) Big data and e-government: A review, in *Information Technology (ICIT), 2017 8th International Conference on*, IEEE, pp. 580–587.

7 Zheng, L., Diao, R., and Shen, Q. (2015) Self-adjusting harmony search-based feature selection. *Soft Computing*, **19** (6), 1567–1579.

8 Zhang, Q., Xiao, Y., Suo, J., Shi, J., Yu, J., Guo, Y., Wang, Y., and Zheng, H. (2017) Sonoelastomics for breast tumor classification: a radiomics approach with clustering-based feature selection on sonoelastography. *Ultrasound in Medicine and Biology*, **43** (5), 1058–1069.

9 Abualigah, L.M., Khader, A.T., and Hanandeh, E.S. (2017) A new feature selection method to improve the document clustering using particle swarm optimization algorithm. *Journal of Computational Science*.

10 Alomari, O.A., Khader, A.T., Al-Betar, M.A., and Abualigah, L.M. (2017) Gene selection for cancer classification by combining minimum redundancy maximum relevancy and bat-inspired algorithm. *International Journal of Data Mining and Bioinformatics*, **19** (1), 32–51.

11 Abualigah, L.M., Khader, A.T., and Hanandeh, E.S. (2019) Modified krill herd algorithm for global numerical optimization problems, in *Advances in Nature-Inspired Computing and Applications*, Springer, pp. 205–221.

12 Dhou, K. (2018) A novel agent-based modeling approach for image coding and lossless compression based on the wolf-sheep predation model, in *International Conference on Computational Science*, Springer, pp. 117–128.

13 Alomari, O.A., Khader, A.T., Mohammed, A.A.B., Abualigah, L.M., Nugroho, H., Chandra, G.R., Katyayani, A., Sandhya, N., Hossain, J., Sani, N.F.M. et al. (2017) Mrmr ba: A hybrid gene selection algorithm for cancer classification. *Journal of Theoretical and Applied Information Technology*, **95** (12).

14 Abualigah, L.M.Q. and Hanandeh, E.S. (2015) Applying genetic algorithms to information retrieval using vector space model. *International Journal of Computer Science, Engineering and Applications*, **5** (1), 19.

15 Abualigah, L.M., Khader, A.T., Al-Betar, M.A., and Alomari, O.A. (2017) Text feature selection with a robust weight scheme and dynamic dimension reduction to text document clustering. *Expert Systems with Applications*, **84**, 24–36.

16 Forsati, R., Mahdavi, M., Shamsfard, M., and Meybodi, M.R. (2013) Efficient stochastic algorithms for document clustering. *Information Sciences*, **220**, 269–291.

17 Geem, Z.W., Kim, J.H., and Loganathan, G. (2001) A new heuristic optimization algorithm: harmony search. *Simulation*, **76** (2), 60–68.

18 Moon, Y.Y., Geem, Z.W., and Han, G.T. (2018) Vanishing point detection for self-driving car using harmony search algorithm. *Swarm and Evolutionary Computation*.

19 Al-Betar, M.A., Khader, A.T., and Zaman, M. (2012) University course timetabling using a hybrid harmony search metaheuristic algorithm. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, **42** (5), 664–681.

20 Reddy, S.S. (2018) Optimal power flow using hybrid differential evolution and harmony search algorithm. *International Journal of Machine Learning and Cybernetics*, pp. 1–15.

21 Al-Betar,M.A. and Khader, A.T. (2012) A harmony search algorithm for university course timetabling. *Annals of Operations Research*, **194** (1), 3–31.

22 Al-Betar, M.A., Khader, A.T., and Liao, I.Y. (2010) A harmony search with multi-pitch adjusting rate for the university course timetabling, in *Recent advances in harmony search algorithm*, Springer, pp. 147–161.

23 Diao, R. (2014) *Feature selection with harmony search and its applications*, Ph.D. thesis, Aberystwyth University.

24 Shamsinejadbabki, P. and Saraee, M. (2012) A new unsupervised feature selection method for text clustering based on genetic algorithms. *Journal of Intelligent Information Systems*, **38** (3), 669–684.

25 Lin, K.C., Zhang, K.Y., Huang, Y.H., Hung, J.C., and Yen, N. (2016) Feature selection based on an improved cat swarm optimization algorithm for big data classification. *The Journal of Supercomputing*, pp. 1–12.

26 Abualigah, L.M., Khader, A.T., AlBetar, M.A., and Hanandeh, E.S. (2017) Unsupervised text feature selection technique based on particle swarm optimization algorithm for improving the text clustering. *EAI Google Scholar*.

27 Alyasseri, Z.A.A., Khader, A.T., Al-Betar, M.A., and Abualigah, L.M. (2017) Ecg signal denoising using β-hill climbing algorithm and wavelet transform, in *Information Technology (ICIT), 2017 8th International Conference on*, IEEE, pp. 96–101.

28 Bolaji, A.L., Al-Betar, M.A., Awadallah, M.A., Khader, A.T., and Abualigah, L.M. (2016) A comprehensive review: Krill herd algorithm (kh) and its applications. *Applied Soft Computing*.

29 Abualigah, L.M., Khader, A.T., Al-Betar, M.A., and Awadallah, M.A. (2016) A krill herd algorithm for efficient text documents clustering, in *Computer Applications & Industrial Electronics (ISCAIE), 2016 IEEE Symposium on*, IEEE, pp. 67–72.

30 Shehab, M., Khader, A.T., Al-Betar, M.A., and Abualigah, L.M. (2017) Hybridizing cuckoo search algorithm with hill climbing for numerical optimization problems, in *Information Technology (ICIT), 2017 8th International Conference on*, IEEE, pp. 36–43.

31 Abualigah, L.M., Khader, A.T., Hanandeh, E.S., and Gandomi, A.H. (2017) A novel hybridization strategy for krill herd algorithm applied to clustering techniques. *Applied Soft Computing*, **60**, 423–435.

32 Abualigah, L.M.Q. (2019) *Feature Selection and Enhanced Krill Herd Algorithm for Text Document Clustering*, Springer.

33 Abualigah, L.M.Q. (2019) Proposed methodology, in *Feature Selection and Enhanced Krill Herd Algorithm for Text Document Clustering*, Springer, pp. 61–103.

34 Abualigah, L.M. and Khader, A.T. (2017) Unsupervised text feature selection technique based on hybrid particle swarm optimization algorithm with genetic operators for the text clustering. *The Journal of Supercomputing*, **73** (11), 4773–4795.

35 Abualigah, L.M., Khader, A.T., and Hanandeh, E.S. (2018) A novel weighting scheme applied to improve the text document clustering techniques, in *Innovative Computing, Optimization and Its Applications*, Springer, pp. 305–320.

36 Bharti, K.K. and Singh, P.K. (2014) A three-stage unsupervised dimension reduction method for text clustering. *Journal of Computational Science*, **5** (2), 156–169.

37 Tsai, C.F., Eberle, W., and Chu, C.Y. (2013) Genetic algorithms in feature and instance selection. *Knowledge-Based Systems*, **39**, 240–247.

38 Mohamad, M.S., Deris, S., Yatim, S., and Othman, M. (2004) Feature selection method using genetic algorithm for the classification of small and high dimension data, in *Proceedings of the 1st International Symposium on Information and Communication Technology*, pp. 1–4.

39 Abualigah, L.M., Khader, A.T., and Al-Betar, M.A. (2016) Unsupervised feature selection technique based on genetic algorithm for improving the text clustering, in *2016 7th International Conference on Computer Science and Information Technology (CSIT)*, pp. 1–6, doi:10.1109/CSIT.2016.7549453.

40 Abualigah, L.M.Q. (2019) Krill herd algorithm, in *Feature Selection and Enhanced Krill Herd Algorithm for Text Document Clustering*, Springer, pp. 11–19.

41 Uğuz, H. (2011) A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm. *Knowledge-Based Systems*, **24** (7), 1024–1032.

42 Zhao, Z., Wang, L., Liu, H., and Ye, J. (2013) On similarity preserving feature selection. *Knowledge and Data Engineering, IEEE Transactions on*, **25** (3), 619–632.

43 Abualigah, L.M., Khader, A.T., and Hanandeh, E.S. A hybrid strategy for krill herd algorithm with harmony search algorithm to improve the data clusterin. *Intelligent Decision Technologies*, (Preprint), 1–12.

44 Rashaideh, H., Sawaie, A., Al-Betar, M.A., Abualigah, L.M., Al-Laham, M.M., Ra'ed, M., and Braik, M. A grey wolf optimizer for text document clustering. *Journal of Intelligent Systems*.

45 Abualigah, L.M., Khader, A.T., and Al-Betar, M.A. (2016) Multi-objectives-based text clustering technique using k-mean algorithm, in *2016 7th International Conference on Computer Science and Information Technology (CSIT)*, pp. 1–6, doi:10.1109/CSIT.2016.7549464.

46 MacQueen, J. et al. (1967) Some methods for classification and analysis of multivariate observations, in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, Oakland, CA, USA., vol. 1, pp. 281–297.

47 Forsati, R., Meybodi, M., Mahdavi, M., and Neiat, A. (2008) Hybridization of k-means and harmony search methods for web page clustering, in *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology-Volume 01*, IEEE Computer Society, pp. 329–335.