N. Karthika* and B. Janet

Feature Pair Index Graph for Clustering

https://doi.org/10.1515/jisys-2018-0338
Received August 14, 2018; previously published online January 15, 2019.

Abstract: Text documents are significant arrangements of various words, while images are significant arrangements of various pixels/features. In addition, text and image data share a similar semantic structural pattern. With reference to this research, the feature pair is defined as a pair of adjacent image features. The innovative feature pair index graph (FPIG) is constructed from the unique feature pair selected, which is constructed using an inverted index structure. The constructed FPIG is helpful in clustering, classifying and retrieving the image data. The proposed FPIG method is validated against the traditional KMeans++, KMeans and Farthest First cluster methods which have the serious drawback of initial centroid selection and local optima. The FPIG method is analyzed using Iris flower image data, and the analysis yields 88% better results than Farthest First and 28.97% better results than conventional KMeans in terms of sum of squared errors. The paper also discusses the scope for further research in the proposed methodology.

Keywords: Inverted index, inverted feature pair index, feature pair index graph, clustering.

1 Introduction

There is extraordinary growth of information on the World Wide Web and users want to preserve documents online. They expect that information must be accessible instantly whenever there is a need. The main aim of the researcher is to organize voluminous information so that it is readily available to the users whenever they need it [11, 31].

The information available significantly relies on the data mining techniques. There are many techniques that reveal the inherent organization of such information/data. Among such methods, clustering is an active approach that identifies the inbuilt grouping of documents where the documents exhibit high intracluster similarity and low intercluster similarity.

In general, the clustering methods discriminate one group of documents from another group of documents where each group deals with data that are different from the other. Clustering reveals the structure and content of unidentified data information and in addition provides a new outlook on familiar ones.

The inspiration behind the proposed method is that clustering should be based on word pairs along with a single term. Word pair indexing in information retrieval systems is used as an additive approach and even shows improvement in precision, but it is done on text data. The review of literature reveals that word pair indexing has not been tried on image data. Incorporation of feature pairs in the feature pair index graph (FPIG) results in a group of similar data. This paper proposes an FPIG for a group of similar data identification which is inspired from text clustering. The primary idea is that the unique feature pair of the image features is represented in the FPIG that yields a group of data which are almost similar to each other. The combination of the following components contributes to identifying the group of similar data, and retrieving and classifying the data:

- The primary idea is to use feature pairs of image data as an essential component.
- An innovative FPIG that captures the feature pairs and clusters the similar data.

***Corresponding author: N. Karthika,** Department of Computer Applications, National Institute of Technology, Tiruchirappalli, Tamil Nadu 620015, India, e-mail: bharathikarthika@gmail.com
B. Janet: Department of Computer Applications, National Institute of Technology, Tiruchirappalli, Tamil Nadu 620015, India

The rest of the paper is organized as follows. In Section 2, we focus on the background study. Section 2.1 describes the preliminaries. Next, in Section 2.2, a semantic correlation between images and text documents is discussed. In addition, a concise view of the overall framework is introduced in Section 3. Details of FPIG construction are given with examples in Section 3.1. Then, in Section 4, the performance of the proposed method is tested against noted machine learning approaches, i.e. KMeans, KMeans++ and Farthest First. Finally, Section 6 summarizes and concludes the work.

2 Background

Index Structures

Information retrieval means recovery of relevant documents from the huge collection of corpus [13]. There are two stages, i.e. indexing and retrieval. Indexing, which represents the content of a document, plays a vital role in efficient retrieval of documents.

Among many index structures, inverted index is dominant in modern search engines. It also outperforms the other index structures in many aspects such as space, speed and functionality for text indexing [32]. Moreover, the efficiency of inverted index has not been surpassed by any other index structures. With reference to [21], mathematical expressions are retrieved when symbol pairs are mapped to expressions they contain. The importance of inverted index in data mining applications is stated in [8].

Various time and space efficient schemes were proposed for information retrieval, but with the exponential increase of information, the conventional index structures became obsolete and the retrieval system's performance was affected in a drastic manner. All these techniques are merely implemented for fast response, i.e. to make retrieval faster. Users browse using bag of words rather than single words. Thus, next word indexes, the consecutive terms will be stored with the position information [4, 20, 25] and two term indexes [22] and word pair indexes have also been proposed [9, 11]. By simply viewing, each pair of terms in a corpus is considered as a single term in the index. It is multiple times faster than the classical index structures.

Though these techniques improve the speed, the space complexity increases. Therefore, many researchers concentrated on compression techniques to reduce storage space [1, 2, 23]. To reduce storage space further, partial next word indexes were proposed [3]. Later, the combination of inverted index, partial phrase index and partial next word index was also introduced to moderately reduce the query time [26].

Document Clustering

On the other hand, professionals concentrated on organizing methodologies of long ordered retrieval results to improve the efficiency of the retrieved results. It considerably reduces the time users spend in searching the elongated ordered list and insight into the documents to identify internal structure of the document. Many of the clustering techniques mostly use a single term [27]. Statistical phrase extraction has become a center of attention due to intensive computational risk in extracting phrases. A consecutive sequence of words is known as the statistical phrase. To extract phrases, the suffix tree model is prominently used [28].

Several clustering methodologies have been introduced based on suffix tree and graph partitioning methods [14]. In word intersection clustering, the word common to all documents in the collection is represented as the center. Similarly, the phrases common to all documents are taken as representative to improve efficiency. The suffix tree implementation of phrase intersection technology is competent enough [17, 29] to overcome the deficiency in numerical algorithms (scatter-gather) [5]. Vivisimo is the commercial implementation of clustering idea (http://www.vivisimo.com). Moreover, an indexing model is documented in [6] based on phrases in addition to trapping the structure of the sentence that was suggested to refrain from high redundancies in the suffix tree model.

Similarities of Text and Image

Similarities of text and image [30] gave deep insight into contextual similarities between text data and image data. The combination of inverted index and spatial information has upgraded image retrieval accuracy [16]. In order to use techniques available in text data for image databases, two steps are followed. The first

step involves obtaining a group of features of image. The second step describes a function which maps from features of image to integers. Through mapping from features of image into cluster, image words are identified, also known as the image bag-of-words model [12].

Based on the above-mentioned concepts, most of the authors tried to upgrade the retrieval process. On the other hand, the inverted feature pair index was not applied on image data to find out data similarity. This paper proposes a unique FPIG for the image data which results in clustering of data objects.

2.1 Preliminaries

2.1.1 Inverted Index

An inverted index is a prominent indexing method as it has an easy structure. It is an effective method in which there is a posting list for each indexed term, and each posting list consists of a document identifier, the number of times the term occurs which means an in-document frequency, and positions where the term appears in the document, i.e. offsets [10].

2.1.2 Inverted Word Pair Index

The inverted index is modified as an inverted word pair index to quickly retrieve data. Every adjacent term is paired instead of using single terms. In comparison with the inverted index structure, usually the inverted word pair has a small posting list due to the reduced existence of most pairs. In addition, inverted word pair index structure approaches have been fruitfully implemented in text document retrieval [9, 11, 26] and document clustering [15].

2.1.3 Document Index Graph (DIG) Model for Text

The DIG model is based on graph theory. It indexes the document on phrases in place of single words with the maintenance of the sentence structure of the documents without storing repeated information [6]. Every node represents a unique word in the document. This typically conveys that the dictionary/vocabulary of the whole documents and the nodes have information about each unique word. If words are adjacent to each other in any of the documents considering the entire documents, then there will be an edge. The edges correspond to the sentence of the document. They also connect the nodes in the same order in which the words are presented in the sentence of the document. It means the dictionary/vocabulary of the entire document with path information is stored in DIG. This usually constructs an inverted index with sentence information.

2.1.4 Terrier

Terrier is a flexible information retrieval platform. Terrier provides various data structures such as document index, direct index, lexicon index and inverted index. It also supports numerous features for indexing as well as retrieval. Many versions are available for download from http://ir.dcs.gla.ac.uk/terrier/ as an open source framework tool [18, 19].

2.2 Semantic Similarity of Text and Images

Images are a significant arrangement of different pixels where documents are also arranged with different words. Feature is equated to word/term, feature pair to word pair and object to sentence. The term-based approach is to retrieve images that have suitable objects by using parts of objects as queries, which is similar

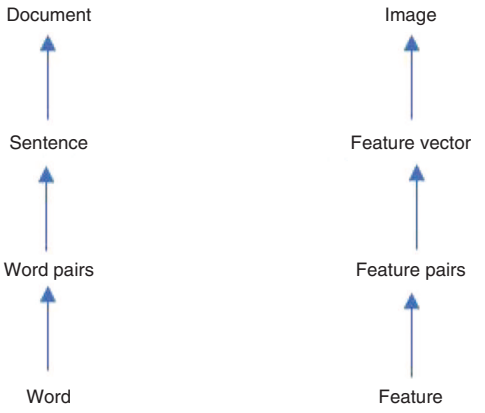

Figure 1: Semantic Comparison of Text and Image.

to retrieval of documents that have suitable sentences by using the terms in the sentences. In these cases, images are seen as a bag of visual features. Furthermore, inverted index is most suitable to index the visual features but still has a false match.

Figure 1 shows the semantic granularities of image and document, i.e. the correspondence between visual features to text words [30]. In a visual feature pair approach which is a simple approach, object feature is granulated into a pair of features, i.e. adjacent features are paired. For these types of approaches, images are viewed as a bag of visual feature phrases. The inverted feature pair index is suitable to index visual feature phrases.

3 Overview of the Proposed Methodology

Image features of the image data are taken as input. The feature vector is constructed and yields a set of unique feature pairs. Then an FPIG is constructed based on the unique feature pairs which provide a group of most similar data. Section 3.1 explains the FPIG process in detail. Figure 2 shows the overall framework.

3.1 Feature Pair Index Graph

This research paper proposes a feature-pair-based approach to retrieve, classify and cluster images. The feature pair is defined as a pair of local adjacent features and is constructed using Terrier 3.5 in the pre-processing step. Furthermore, the FPIG was devised to make retrieval as well as clustering easier. The experiment is more efficient and effective than a feature-based approach. The researcher finds the feature pair index itself to be helpful in clustering. This will be as good as clusters created using the KMeans approach.

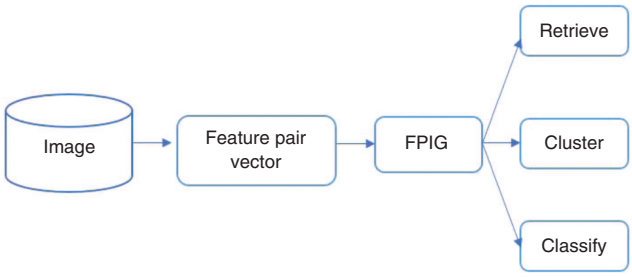

Figure 2: The Overall Framework of the Proposed Work.

3.1.1 FPIG Construction

The FPIG is a directed graph (digraph) G = (V, E). Here V is the set of nodes $\{v_1, v_2, v_3, \ldots, v_n\}$ and each node v_i represents a unique feature pair in the data object. As an example, in Table 1, the feature pair of data object number 1 is {6 2, 2 4, 4 1}. E is the set of edges $\{e_1, e_2, e_3, \ldots, e_m\}$ and there will be an edge e_j from v_k to v_l if and only if the feature pair v_l appears subsequent to the feature pair v_k. For an illustration, in the table the node v_1, i.e. 6 2, and the node v_2, i.e. 2 4, are adjacent in data object number 1. Hence, an edge e_1 is created between those nodes.

The above description conveys that the number of nodes in the graph is the number of unique feature pairs.

Table 2 shows the details of edge construction between nodes in the FPIG and Figure 3 is a complete representation of a sample of Iris data which are shown in Table 1 and the corresponding nodes in Table 3. As described, In the FPIG, each node represents the unique feature pair. An edge is created between two nodes only when the feature pair occurs successively in the object.

The example given in Figure 3 shows that nodes 6 2 and 2 4 occur adjacently. Hence, an edge is created between these nodes. The feature pair which occurs commonly is shared in the graph. For example, if we

Table 1: Sample Data.

Data object no.	Class	Iris data
1	2	6 2 4 1
2	3	7 3 5 2
3	2	5 2 4 1
4	3	7 3 6 2

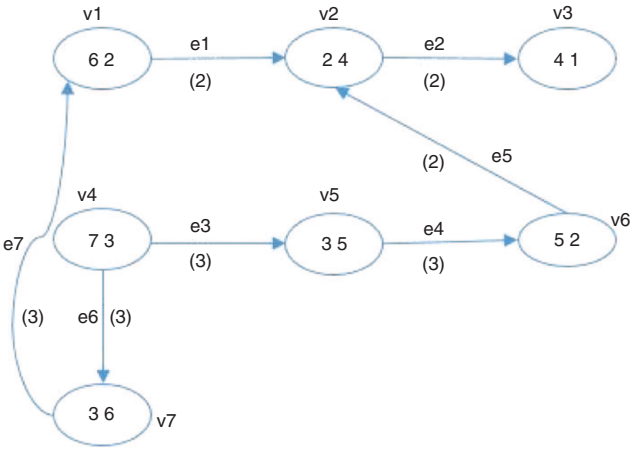

Figure 3: Feature Pair Index Graph.

Table 2: Edge Table of Sample Data.

Graph G			
Edge	**vi**	**vj**	**Class**
e1	v1	v2	2
e2	v2	v3	2
e3	v4	v5	3
e4	v5	v6	3
e5	v6	v2	2
e6	v4	v7	3
e7	v7	v1	3

Table 3: Nodes and Node Id of Sample Data.

List L	
Nodes	**Node Id**
6 2	v1
2 4	v2
4 1	v3
7 3	v4
3 5	v5
5 2	v6
3 6	v7

consider node 4 1, it can be traversed through the path 6 2, 2 4 and again through the path 5 2, 2 4. In this case, the paths get shared & class updated according without creating a new path.

Algorithm 1: FPIG Construction.

Input: V:v_i $(1 < i < n)$-Set of Nodes
Output: G-Graph
1: L $\longleftarrow$ Empty node list
2: v_1 $\longleftarrow$ first node in V
3: **if** $v_1 \notin$ G **then**
4: Add v_1 into G and add into L
5: **end if**
6: **for** each v_i $(2 < i < n) \in$ V **do**
7: **if** $\langle v_i, v_{i+1} \rangle$ **then**
8: Add an edge e_j $(1 < j < n)$ in G and add v_{i+1} into L
9: **end if**
10: **end for**

The process starts with an empty list L (line 1). v_i is added from the set of nodes as the first node in the graph (lines 2–5). Lines 6–10 explain about that next node to be read. If these are adjacent to each other, they establish an edge between them and update list L. This process continues until all the nodes are read.

4 Experimental Measures and Metrics

The quality of a clustering solution was ensured by using the following metrics, shown in Table 4.

Table 4: Evaluation Metrics.

Measures	Formulae
Mean	$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x$ (x is a data point)
Variance	$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x-\bar{x})^2$
Standard deviation	$s^2 = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x-\bar{x})^2}$
Euclidean distance	$\sqrt{\sum_{i=1}^{n}(q_i - p_i)^2}$ (p & q are data points)
SSE (sum of squared errors)	$\sum_{i=1}^{n}(x_i - \bar{x})^2$
SICD (sum of intracluster distances)	$\sum_{i=1}^{n}\sum_{j=1}^{m}(x_j - \bar{x})^2$ (n is the number of clusters, m is a data point)

5 Results and Discussion

The proposed framework is evaluated on the Iris flower data set which is downloadable from UCI machine learning laboratory [24] using 7th Generation Intel core i7 processor, 16 GB of RAM, 1 TB HDD for windows and Java Run Time Environment of version 1.8.0_151.

The sum of squared errors (SSE) is used as the primary quality measure; the lesser the errors, higher the quality. The *x*-axis in Figure 4 has the various approaches examined against the proposed FPIG wherein the *y*-axis shows the SSE. The analysis of the graph plotted shows the higher error rate in Farthest First. In comparison with Farthest First, the innovative FPIG has 88% lesser errors. In addition, the FPIG achieved 28.97% better results than KMeans, while KMeans++ yielded 45.96% better results than the FPIG. KMeans++ is 61.62% better than KMeans. Farthest First is 83.79% better than KMeans. The proposed FPIG was examined against the three traditional methods. Out of the three, the proposed FPIG was superior to two approaches.

Table 5 summarizes the statistical details of the examined approaches. Variance shows how data are spread out and the spanning range of values that the methods formed.

Table 6 summarizes the intracluster distances attained from the FPIG and the traditional KMeans, Farthest First and KMeans++ methods. From this table, it is perceived that the proposed FPIG has the higher intracluster distance. Although the proposed FPIG achieved best SSE, i.e. lower error rate, it did not result in lesser sum of intracluster distances. It should be perceived that the intracluster distance is not proportional to error rate [7] and there is no relationship between intracluster distance and the error rate. The Farthest First

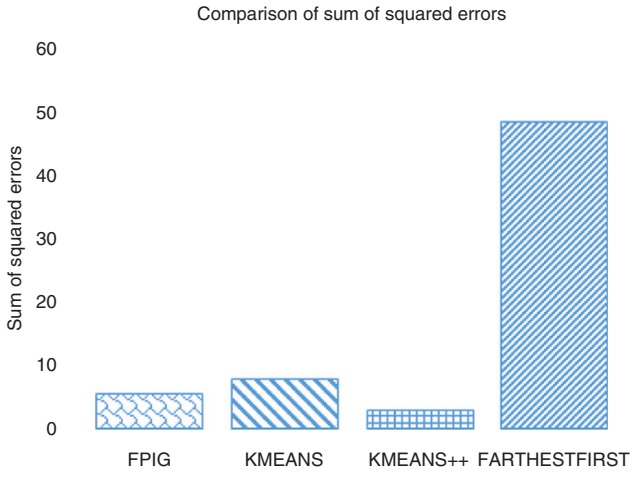

Figure 4: Sum of Squared Errors Between the FPIG and the Traditional Methods.

Table 5: Statistical Values of Examined Methods.

Methods	Mean	Standard deviation	Variance
KMeans	3.566	0.44	0.210
KMeans++	3.37	0.41	0.196
Farthest First	3.47	1.001	1.254
FPIG	3.767	1.373	2.16

Table 6: Sum of Intracluster Distances Between Analyzed Methods.

Approaches	Sum of intra cluster distances
KMeans	1034
KMeans++	612
Farthest First	1095
FPIG	1552

Table 7: Time Taken to Accomplish the Methods.

Approaches	Time taken (ms)
KMeans	1788
KMeans++	8375
Farthest First	10,102
FPIG	18,000

achieved 29%, the KMeans obtained 33% and KMeans++ achieved 60% better results than the proposed FPIG. KMeans++ is 40.81% better than KMeans and 44.10% better than the Farthest First method.

Table 7 shows the time taken to accomplish the task by various methods. KMeans took 78.65% lesser than KMeans++, 82.30% lesser than Farthest First and 90.06% lesser than the FPIG. Farthest First took 43.87% lesser than the FPIG. KMeans++ took 17.09% lesser than the Farthest First method.

There is further research scope for the innovative FPIG approach identified:

- The application of this approach on various other kinds of data type.
- Time complexity of the innovative methodology gives a few more directions to analyze.

6 Conclusion and Future Scope

This paper has proposed an innovative FPIG to retrieve, classify and cluster the image information. The performance of the FPIG method is examined in terms of SSE against conventional clustering approaches such as KMeans, KMeans++ and Farthest First. The proposed FPIG yields 88% lower SSE than Farthest First and 28.97% lesser SSE than KMeans. Out of the three methods examined against the proposed FPIG, the innovative approach performance is superior to two approaches. Furthermore, the proposed methodology will be considered for the other multimedia data.

Bibliography

[1] V. N. Anh and A. Moffat, Inverted index compression using word-aligned binary codes, *Inform. Retrieval* **8** (2005), 151–166.
[2] D. Bahle, H. E. Williams and J. Zobel, Compaction techniques for nextword indexes, in: *Spire*, pp. 33–45, IEEE, Hoboken, NJ, USA, 2001.
[3] D. Bahle, H. E. Williams and J. Zobel, Efficient phrase querying with an auxiliary index, in: *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 215–221, ACM, Tampere, Finald, 2002.
[4] M. Chang and C. K. Poon, Efficient phrase querying with common phrase index, in: *European Conference on Information Retrieval*, pp. 61–71, Springer, London, UK, 2006.
[5] D. R. Cutting, D. R. Karger, J. O. Pedersen and J. W. Tukey, Scatter/gather: a cluster-based approach to browsing large document collections, in: *ACM SIGIR Forum*, Vol. 51, pp. 148–159, ACM, Tokyo, Japan, 2017.
[6] K. M. Hammouda and M. S. Kamel, Efficient phrase-based document indexing for web document clustering, *IEEE Trans. Knowl. Data Eng.* **16** (2004), 1279–1296.
[7] X. Han, L. Quan, X. Xiong, M. Almeter, J. Xiang and Y. Lan, A novel data clustering algorithm based on modified gravitational search algorithm, *Eng. Appl. Artif. Intell.* **61** (2017), 1–7.
[8] M. Ilic, P. Spalevic and M. Veinovic, Inverted index search in data mining, in: *Telecommunications Forum Telfor (TELFOR)*, pp. 943–946, IEEE, Belgrade, Serbia, 2014.
[9] B. Janet and A. Reddy, Cube index for unstructured text analysis and mining, in: *Proceedings of the 2011 International Conference on Communication, Computing & Security*, pp. 397–402, ACM, Odisha, India, 2011.
[10] W. Jung, H. Roh, M. Shin and S. Park, Inverted index maintenance strategy for flashSSDs: revitalization of in-place index update strategy, *Inform. Systems* **49** (2015), 25–39.
[11] N. Karthika and B. Janet, Word pair index structure for information retrieval using Terrier 3.5, in: *IEEE International Conference on Computational Intelligence in Data Science (ICCIDS)*, 2017.
[12] A. Ma, A. Flenner, D. Needell and A. G. Percus, Improving image clustering using sparse text and the wisdom of the crowds, in: *48th Asilomar Conference on Signals, Systems and Computers*, pp. 1555–1557, IEEE, Pacific Grove, CA, USA, 2014.

[13] C. D. Manning, P. Raghavan and H. Schütze, *Introduction to information retrieval*, Vol. 1, Cambridge University Press, Cambridge, 2008.
[14] I. Masłowska, Phrase-based hierarchical clustering of web search results, in: *European Conference on Information Retrieval*, pp. 555–562, Springer, Pisa, Italy, 2003.
[15] B. Momin, P. Kulkarni and A. Chaudhari, Web document clustering using document index graph, in: *International Conference on Advanced Computing and Communications (ADCOM)*, pp. 32–37, IEEE, Surathkal, India, 2006.
[16] V. T. Nguyen, T. D. Ngo, M. T. Tran, D. D. Le and D. A. Duong, A combination of spatial pyramid and inverted index for large-scale image retrieval, *Int. J. Multimedia Data Eng. Manage.* **6** (2015), 37–51.
[17] S. Osiński, J. Stefanowski and D. Weiss, Lingo: search results clustering algorithm based on singular value decomposition, in: *Intelligent Information Processing and Web Mining*, pp. 359–368, Springer, Zakopane, Poland, 2004.
[18] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald and D. Johnson, Terrier information retrieval platform, in: *European Conference on Information Retrieval*, pp. 517–519, Springer, Gdansk, Poland, 2005.
[19] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald and C. Lioma, Terrier: a high performance and scalable information retrieval platform, in: *Proceedings of the OSIR Workshop*, pp. 18–25, Seattle, WA, USA, 2006.
[20] M. Patil, S. V. Thankachan, R. Shah, W. K. Hon, J. S. Vitter and S. Chandrasekaran, Inverted indexes for phrases and strings, in: *Proceedings of the 34th international ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 555–564, ACM, Beijing, China, 2011.
[21] D. Stalnaker and R. Zanibbi, Math expression retrieval using an inverted index over symbol pairs, in: *Document Recognition and Retrieval XXII*, Vol. 9402, p. 940207, International Society for Optics and Photonics, 2015.
[22] F. Transier and P. Sanders, Out of the box phrase indexing, in: *International Symposium on String Processing and Information Retrieval*, pp. 200–211, Springer, Melbourne, Australia, 2008.
[23] A. Trotman, Compressing inverted files, *Inform. Retrieval* **6** (2003), 5–19.
[24] UCI. https://archive.ics.uci.edu/ml/, Accessed July, 2018.
[25] H. E. Williams, J. Zobel and P. Anderson, What's next? Index structures for efficient phrase querying, in: *Australasian Database Conference*, pp. 141–152, Auckland, New Zealand, 1999.
[26] H. E. Williams, J. Zobel and D. Bahle, Fast phrase querying with combined indexes, *ACM Trans. Inform. Syst.* **22** (2004), 573–594.
[27] D. Xu and Y. Tian, A comprehensive survey of clustering algorithms, *Ann. Data Sci.* **2** (2015), 165–193.
[28] O. Zamir and O. Etzioni, Web document clustering: a feasibility demonstration, in: *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 46–54, ACM, Melbourne, Australia, 1998.
[29] O. Zamir, O. Etzioni, O. Madani and R. M. Karp, Fast and intuitive clustering of web documents, in: *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, pp. 287–290, AAAI Press, Newport Beach, CA, USA, 1997.
[30] Q. F. Zheng and W. Gao, Constructing visual phrases for effective and efficient object-based image retrieval, *ACM Trans. Multimedia Comput. Commun. Appl.* **5** (2008), 7.
[31] J. Zobel and A. Moffat, Inverted files for text search engines, *ACM Comput. Surv.* **38** (2006), 6.
[32] J. Zobel, A. Moffat and K. Ramamohanarao, Inverted files versus signature files for text indexing, *ACM Trans. Database Syst.* **23** (1998), 453–490.