Global Journal of Computer Science and Technology: C
Software & Data Engineering
Volume 16 Issue 5 Version 1.0 Year 2016
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

Feature Extraction and Duplicate Detection for Text Mining: A Survey

By Ramya R S, Venugopal K R, Iyengar S S & Patnaik L

University Visvesvaraya College of Engineering

Abstract- Text mining, also known as Intelligent Text Analysis is an important research area. It is very difficult to focus on the most appropriate information due to the high dimensionality of data. Feature Extraction is one of the important techniques in data reduction to discover the most important features. Proce- ssing massive amount of data stored in a unstructured form is a challenging task. Several pre-processing methods and algo- rithms are needed to extract useful features from huge amount of data. The survey covers different text summarization, classi- fication, clustering methods to discover useful features and also discovering query facets which are multiple groups of words or phrases that explain and summarize the content covered by a query thereby reducing time taken by the user.

Keywords: text feature extraction, text mining, query search, text classification.

GJCST-C Classification: C.2.1,C.2.4,H.2.8

Feature Extraction and Duplicate Detection for Text Mining: A Survey

Ramya R S [α], Venugopal K R [σ], Iyengar S S [ρ] & Patnaik L M [ω]

Abstract- Text mining, also known as Intelligent Text Analysis is an important research area. It is very difficult to focus on the most appropriate information due to the high dimensionality of data. Feature Extraction is one of the important techniques in data reduction to discover the most important features. Processing massive amount of data stored in a unstructured form is a challenging task. Several pre-processing methods and algorithms are needed to extract useful features from huge amount of data. The survey covers different text summarization, classification, clustering methods to discover useful features and also discovering query facets which are multiple groups of words or phrases that explain and summarize the content covered by a query thereby reducing time taken by the user. Dealing with collection of text documents, it is also very important to filter out duplicate data. Once duplicates are deleted, it is recommended to replace the removed duplicates. Hence we also review the literature on duplicate detection and data fusion (remove and replace duplicates).The survey provides existing text mining techniques to extract relevant features, detect duplicates and to replace the duplicate data to get fine grained knowledge to the user.

Keywords: text feature extraction, text mining, query search, text classification.

I. Introduction

Society is increasingly becoming more digitized and as a result organisations are producing and storing vast amount of data. Managing and gaining insights from the produced data is a challenge and key to competitive advantage. Web based social applications like people connecting websites results in huge amount of unstructured text data. These huge data contains a lot of useful information. People hardly bother about the correctness of grammar while forming a sentence that may lead to lexical syntactical and semantic ambiguities. The ability of finding patterns from unstructured form of text data is a difficult task.

Data mining aims to discover previously unknown interrelations among apparently unrelated attributes of data sets by applying methods from several areas including machine learning, database systems, and statistics. Many researches have emphasized on different branches of data mining such as opinion mining, web mining, text mining. Text mining is one of the most important strategy involved in the phenomenon of knowledge discovery. It is a technique of selecting previously unknown, hidden, understandable, interesting knowledge or patterns which are not structured. The prime objective of text mining is to diminish the effort made by the users to obtain appropriate information from the collection of text sources [1].

Thus, our focus is on methods that extract useful patterns from texts in order to categorize or structure text collections. Generally, around 80 percent of company's information is saved in text documents. Hence text mining has a higher economic value than data mining. Current research in the area of text mining tackles problems of text representation, classification, clustering, information extraction or the search for and modelling of hidden patterns. Selection of characteristics, influence of domain knowledge and domain-specific procedures play an important role.

The text documents contain large scale terms, patterns and duplicate lists. Queries submitted by the user on web search are usually listed on top retrieved documents. Finding the best query facet and how to effectively use large scale patterns remains a hard problem in text mining. However, the traditional feature selection methods are not effective for selecting text features for solving relevance issue. These issues suggests that we need an efficient and effective methods to mine fine grained knowledge from the huge amount of text documents and helps the user to get information quickly about a user query without browsing tens of pages.

The paper provides a review of an innovative techniques for extracting and classifying terms and patterns. A user query is usually presented in list styles and repeated many times among top retrieved documents. To aggregate frequent lists within the top search results, various navigational techniques have been presented to mine query facets.

The Organisation of the paper is as follows: Section 1 introduces a detailed overview of text mining frameworks, application and benefits of text mining. Sections 2 and 3 reviews feature selection, feature extraction and techniques of pattern extraction. Section 4 discusses various text classification and clustering algorithms in text mining. Sections 5 and 6 introduce a detailed overview of discovering facets and fine grained knowledge. Section 7 reviews the duplicate detection in text documents. Section 8 contains the conclusions.

a) Text Mining Models

Text mining tasks consists of three steps: text preprocessing, text mining operations, text post processing. Text preprocessing includes data selection,

Author α σ ρ ω: Department of Computer Science and Engineering University Visvesvaraya College of Engineering, Bangalore University, Bangalore 560 001. e-mail: rs.ramya.reddy@gmail.com

text categorization and feature extraction. Text mining operations are the core part of text mining that includes association rule discovery, text clustering and pattern discovery as shown in Figure1. Post processing tasks modifies the data after text mining operations are completed such as selecting, evaluating and visualization of knowledge. It consists of two components text filtering and knowledge cleansing. Many approaches [2] have been concerned of obtaining structured datasets called intermediate forms, on which techniques of data mining [3] are executed.

Text filtering translates collection of text documents into selected intermediate form(IF) which means Knowledge cleansing or discovering patterns. It can be structured or semistructured. Text mining methods like clustering, cla- ssification and feature extraction falls within document based IF. Pattern discovery and relationship of the obje- ct, associative discovery, visualization fall within object based documents.

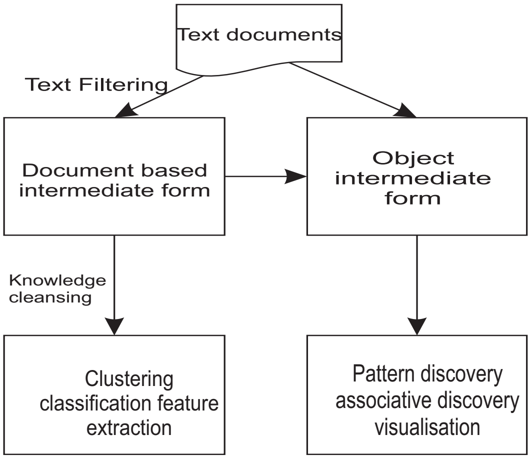

Text documents
Text Filtering
Document based intermediate form
Object intermediate form
Knowledge cleansing
Clustering classification feature extraction
Pattern discovery associative discovery visualisation

Figure 1: Text mining framework

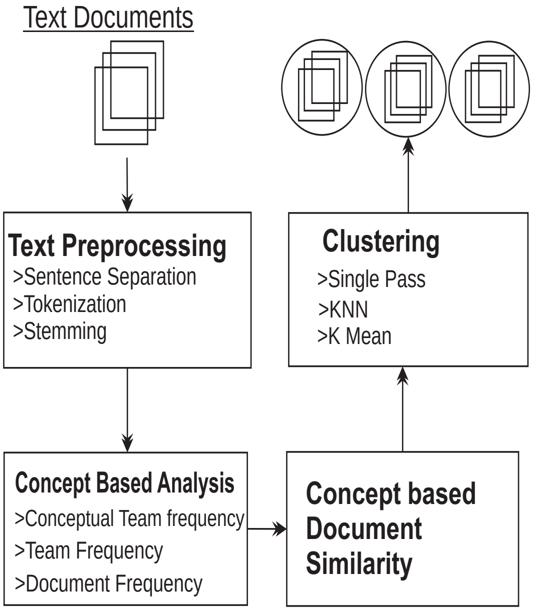


Figure 2: Concept based Mining Model System

When documents contains terms with same frequency. Two terms can be meaningful while the other term may be irrelevant. Inorder to discover the semantic of text, the mining model is introduced. Figure 2 represents a new mining model based on concepts. The model is proposed to analyse terms in a sentence from documents. The model contains group of concept analysis, they are sentence based concept analysis, document based concept analysis and corpus based similarity measure [4]. Similarity measure concept based analysis calculates the similarity between documents. The model effectively and efficiently finds matching concepts between documents, according to the meaning of their sentence.

b) *Application and benefits of Text Mining*

Text mining [5] [6] [7] has several applications in discovering hidden knowledge and it can be used in one of the three ways.

Clustering: The process of grouping similar kind of information is called clustering that results in finding interesting knowledge. The new discovered knowledge can be used by an industry for further development and helps in competing with their competitors.

Question Answering: For seperating and combining terms we use standard text searching techniques that use boolean operators. Sophisticated search in text mining executes the searching process in sentence or phrase level and verbal connection identification between various search terms, which is not possible in traditional search. The result obtained by sophisticated search can be used for providing specific information that can be influenced by an organization.

Concept linkage: The results obtained from sophisticated search are linked together to produce a new hypothesis. The linking of concepts is called concept linkage. Hence, new domain of knowledge can be generated by making use of concept linkage application.

Benefits of text mining are better collection development to resolve user needs, information retrieval, to resolve usability and system performance, data base evaluation, hypothesis development. Information professionals(IP) [8] are always in forefront for emerging technologies. Inorder to make their product and service better and more efficient, usually libraries and information use these IP. The trained information professionals manage both technical and semantic infrastructures which is very important in text mining. IP also manages content selection and formulation of search techniques and algorithms.

Akilan et al., [9] pesented the challenges and future directions in text mining. It is mandatory to function semantic analysis to capture objects relationship in the documents. Semantic analysis is computationally expensive and operates on few words per second as text mining consists of significant language component. An effective text refining method has to be developed to process multilingual text document. Trained knowledge specialists are neceessary to deal with products and application of current text mining tools. Automated mining operations is required which can be used by technical users. Domain Knowledge plays an important role in both at text refining stage and knowledge distillation and hence helps in improving the efficiency of text mining.

Sanchez et al., [10] presented Text knowledge mining (TKM) based deductive inference that is usually targeted on the feasible subset of texts which usually search for contradictions. The procedure obtains new knowledge making a union of intermediate forms of texts from accurate knowledge expressed in the text.

Dai et al., [11] introduced competitive intelligence analysis methods FFA (Five Faces Frame work) and SWOT with text mining technologies. The knowledge is extracted from the raw data while performing transforming process that enables the business enterprises to take decisions more reliably and easily. Mining Environment for Decisions (MinEDec) system is not evaluated in real business environments.

Hu et al., [12] presented a interesting task of automatically generating presentation slides for academic papers. Using a support vector regression method, importance scores of sentences in the academic papers is provided. Another method called Integer Linear Programming is used to generate well structured slides. The method provides the researchers to prepare draft slides which helps in final slides used for presentation. The approach does not focus on tables, graphs and figures in the academic papers.

c) *Traffic based Event in Text Mining*

Andrea et al., [13] [14] have proposed a real-time monitoring system for traffic event detection that fetches tweets, classifies and then notifies the users about traffic events. Tweets are fetched using some text mining techniques. It provides the class labels to each tweet that are related to a traffic event. If the event is due to an external cause, such as football match, procession and manifestation, the system also discriminate the traffic event. Final result shows it is capable of detecting traffic event but traffic condition notifications in real-time is not captured.

An efficient and scalable system from a set of microblogs/ tweets has been proposed to detect Events from Tweets (ET) [15] by considering their textual and temporal components. The main goal of proposed ET system is the efficient use of content similarity and appearance similarity among keywords and to cluster the related keywords. Hierarchical clustering technique is used to determine the events, which is based on common co-occurring features of keywords [16]. ET is evaluated on two different datasets from two different domains. The results show that it is possible to detect events of relevance efficiently. The use of semantic knowledge base like Yago is not incorporated.

Schulz et al., [17] proposed a machine learning algorithm which includes text classification and increasing the semantics of the microblog. It identifies the small scale incidents with high accuracy. It also precisely localizes microblogs in space and time which enables it to detect incidents in real time. The algorithm will not only give us information about the incident and in addition give us valuable information on previous unknown information about the incidents. It does not considers NLP techniques and large data.

ITS (Intelligent Transportation Systems) [18] recognizes the traffic panels and dig in information contained on them. Firstly, it applies white and blue color segmentation and then at some point of interest it derives descriptors. These images that can now be considered as sack of words and classified using Na¨ıve Bayes or SVM (state vector method). The kind of categorization where the images are classified based on visual appearance is new for traffic panel detection and it does not recognize multiframe integration.

II. Preprocessing in Text Mining

Text may be loosely organized without complete information in the documents and may also contain omitted information. The text has to be scanned attentively to determine the problems. If it is not scanned and scrutinised properly then it leads to poor accuracy on unstructured data and hence preprocessing is necessary.

Preprocessing guarantees successful implementation of text analysis, but may spend substantial processing time. Text processing can be done in two basic methods. a)Feature Selection b) Feature Extraction.

a) Feature Selection

Research in numerous fields like machine learning, data mining, computer vision, statistics and linked fields has led to diversity of feature selection approaches in supervised and unsupervised surroundings.

Feature Selection (FS) has an important role in data mining in categorization of text. The centralized idea of feature selection is the reduction of the dimension of the feature set by determining the features appropriately which enhances the efficiency and the performance. FS is a search process and categorized into forward search and backward search.

Mehdi et al., [19] [20] executed a innovative feature selection algorithm based on Ant Colony Optimization (ACO).

Without any prior knowledge of features, a minimal feature subset is determined by applying ACO [21]. The approach uses simple nearest neighbor classifier to show the effectiveness of ACO algorithm by reducing the computational cost and it outperforms information gain and chi methods. Complex classifiers and different kinds of datasets are not incorporated. Combining feature selection algorithm with other population-based feature selection algorithms are not considered.

Gasca et al., [22] proposed feature selection method based on Multilayer Perceptron (MLP). Under certain objective functions the approach determines and also corrects proper set of irrelevant set of attributes. It further computes the relative contributions for individual attribute in reference to the units that are to be output. For each output unit, contribution are sorted in the descending order. An objective function called prominance is computed for each attribute. Selecting the features from large document faces problem in unsupervised learning because of unnamed class labels.

Sivagaminathan et al., [23] [24] proposed a fixed size subset, an hybrid approach to solve feature subset selection problem in neural network pattern classifier. It considers both the individual performance and subset performance. Features are selected using the pheromone trail and value of heuristic by state transition rules. After selecting the feature, the global updating rule takes place to increment the features, which ultimately gives better classification performance without increase in the overall computational cost.
selection algorithms.

Table 1: shows comparison of feature

Sl.no.	Authors	Feature Selection (FS)	Algorithm	Advantages	Disadvantages
1.	Zhao *et al.*,(2016) [25]	Unsupervised	Gradient	Preserve similarity and discriminant information, high clustering performance is achieved	Supervised FS is not considered
2.	Xu *et al.*,(2016) [26]	-	Deep Learning	Performs better than traditional dimensional reduction method	Meta data information of tweet is not considered
3.	Wang *et al.*,(2015) [27]	Supervised and Unsupervised	Global redundancy Minimization	Features are more compact and discriminant,Superior performance without parameter	-

Ogura et al., [28] proposed an approach to reduce a feature dimension space which calculates the probability distribution for each term that deviates from poissons. These deviations from poissons are non significant for the documents that does not belong to category. Three measures are employed as a benchmark and by using two classifiers SVM and K-NN gives better performance than other conventional classifiers. Gini index proved to be better than chisquare, IG in terms of macro, micro average values of F1. These measures do not utilize the number of times the term occurs in a document. The computational complexity could not be to suppressed for other typical measures such as information gain and CHI.

Feature selection is measured based on words term and document frequency. Azam et al., [29] observes these frequencies for measuring FS. The metrics of Discriminative Power Measure (DPM) and GINI index (GINI) are incorporated and the term frequency based metric is useful for small feature set. The most important features returned by DPM and GINI tend to discover most of the available information at a faster rate, i.e. against lower number of features. The DPM and GINI are comparatively slower in covering document frequency information.

Yan et al., [30] presented a graph embedded framework for dimensionality reduction. The framework is also used as a tool and unifies many feature extraction methods. Feature is selected based on spectral graph theory and proposed framework unifys both supervised and unsupervised feature selection.

Zhao et al., [31] developed a framework for preserving feature selection similarity to handle redundant feature. A combined optimization formulation of sparse multiple output regression formulation is used for selecting similarity preserving features. The framework do not address existing kernel, metric learning methods and semi-supervise feature selection methods.

1) *Feature Selection based Graph Reconstruction:*A Major task in efficient data mining is Feature selection. Feature selection has a significant challenge in small labeled-sample problem. If data is unlabeled then it is large. If the label of data is extremely tiny, then supervised feature selection algorithms fail for want of sufficient information.

Zhao et al., [32] introduced graph regularized data construction to overcome the problems in feature selection. The approach achieves higher clustering performance in both unsupervised and supervised feature selection.

Linked social media crops enormous amount of unlabeled data. In the prevailing system, selecting features for unlabeled data is a difficult task due to the lack of label information. Tang et al., [33] proposed an unsupervised feature selection framework, LUFS(Linked Unsupervised Feature Selection), for related social media data to surpass the problem. The design builds a pseudo-class labels through social dimension extraction and spectral analysis. LUFS efficiently exploits association information but does not exploit link information. Computer vision and pattern recognition problems are the two main problems which have inherent manifold structure. A laplacian regularizer is included to smoothen the clustering process along with the scale factor. In text mining applications, several existing systems incorporate a NLP-basedtechniques which parse the text and promote the usage patterns that is used for mining and examination of the parse trees that are trivial and complex.

Mousavi et al., [34] have formulated a weighted graph depiction of text, called Text Graphs that further captures grammar which serve as semantic dealings between words that are in textual terms. The text based graphs incorporates such a framework called SemScape that creates parse trees for each sentence and uses two step pattern based procedure for facilitation of extraction from parse trees candidate terms and their parsable grammar.

Due to the absence of label information, it is hard to select the discriminative features in unsupervised learning. In the prevailing system, unsupervised feature selection algorithms frequently select the features that preserve the best data dissemination. Yang et al., [35] proposed a new approach that is L2, 1 -norm regularized Unsupervised Discriminative Feature Selection (UDFS). The algorithm chooses the most discriminative feature subset from the entire feature set in batch mode. UDFS outclasses the existing unsupervised feature selection algorithms and selects discriminative features for data representation. The performance is sensitive to the number of selected features and is data dependent.

Cai et al., [36] presented a novel algorithm, called Graph regularized Nonnegative Matrix Factorization (GNMF) [37], which explicitly considers the local invariance. In GNMF, the geometrical information of the data space is pre-arranged by building a nearest neighbor graph and gathering parts-based representation space in which two data points are adequately close to each other, if they are connected in the graph. GNMF models the data space as a sub manifold rooted in the ambient space and achieves more discriminating power than the ordinary NMF approach.

Fan et al., [38] suggested a principled vibrational framework for unsupervised feature selection using the non Gaussian data which is subjective to several applications that range from several diversified domains to disciplines. The vibrational frameworks provides a deterministic alternative for Bayesian approximation by the maximization of a lower bound on the marginal probability which has an advantage of computational efficiency.

2) Text summarization and Dataset: Several approaches have been developed till date for automatic summarization by identifying important topic from single document or clustered documents. Gupta et al., [39] describes a topic representation approach that captures the topic and frequency driven approach using word probability which gives reasonable performance and conceptual simplicity.

Negi et al., [40] developed a system that summarizes the information from a clump of documents. The proposed system constructs the identifiers that are useful for retrieving the important information from the given text. It achieves high accuracy but cannot calculate the relevance of the document.

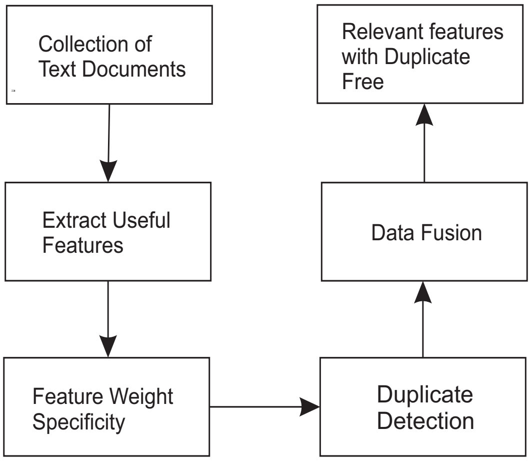
Debole et al., [41] initially explains the three phases in the life cycle of TC system like document indexing, classifier learning and classifier evaluation. All researches takes Reuters 21578 documents for TC experiments. Several researches have used Modapte split for testing. The three subsets used for the experiments are a set of ten categories with more number of positive training examples.

Xie et al., [42] proposed an approach to the acquisition of the semantic features within phrases from a single document that extracts document keyphrases. Keyphrase extraction method always performs better than TFIDF and KEA. Keyphrase extraction is a basic research in text mining and natural language processing. The method is developed on the concept of semantic relatedness where degrees between phrases are calculated by the cooccurrences between phrases in a given document and the same is presented as a relatedness graph. The approach is not domain specific and generalizes well on journal articles and is tested on news web pages.

To obtain any online information is an easy task. We log on to the world wide web and give simple keywords. However, it is not easy for the user to read the entire information provided. Hence text summarization is needed.

b) Feature Extraction

1) *Feature Mining for Text Mining:* Li et al.,[43] designed a new technique to discover patterns i.e., positive and negative in text document. Both relevant and irrelevant document contains useful features. Inorder to remove the noise, negative documents in the training set is used to improve the effectiveness of Pattern Taxonomy Model PTM. Two algorithms HLF mining and N revision was introduced. In HLF mining, it first finds positive features, discovers negative features and then composes the set of term. The offenders are selected by ranking the negative documents. The weights are initialized to the discovered terms of negative patterns. NRevision algorithms explains the terms weight based on their specificity and distribution in both positive and negative patterns.


Collection of Text Documents
Extract Useful Features
Feature Weight Specificity
Duplicate Detection
Data Fusion
Relevant features with Duplicate Free

Zhong et al., [44] has presented an effective pattern discovery technique which includes the process of pattern deploying and pattern evolving as shown in Table 2, to improve the effectiveness of using and updating discovered patterns for finding relevant and interesting information. The proposed model outperforms other pure data mining-based methods, the concept based models and term-based state-of the- art models, such as BM25 and SVM.

Li et al., [47] proposed two algorithms namely Fclustering and Wfeature to discover both positive and negative patterns in the text documents. The algorithm Fclustering classifies the terms into three categories general, positive, negative automatically without using parameters manually. After classifying the terms using Fclustering, Wfeature is executed to calculate the weights of the term. Wfeature is effective because the selected terms size is less than the average size of the documents. The proposed model is evaluated on RCV, Trec topics and Reuters 21578 dataset as shown in Table 2, the model performs much better than the term based method and pattern based method. The use of irrelevance feedback strategy is highly efficient for improving the overall performance of relevance feature discovery model.

Xu et al., [26] experimented on microblog dimensionality reduction- A deep learning approach. The approach aims at extracting useful information from large amount of textual data produced by microblogging services. The approach involves mapping of natural language texts into proper numerical representations which is a challenging issue. Two types of approaches namely modifying training data and modifying training objective of deep networks are presented to use micro-blog specific information. Meta-information contained in tweets like embedded hyperlinks is not explored.

Nguyen et al., [49] worked on review selection using Micro-reviews. The approach consists of two steps namely matching review sentences with micro reviews and selecting a few reviews which cover many reviews. A heuristic algorithm performs computionally fast and provides informative reviews.

2) *Feature Extraction for Classification:* Khadhim et al., [50] [21] developed two weighting methods TF -IDF and TF-IDF (Term Frequency/Inverse Document Frequency) global to reduce dimensionality of datasets because it is very difficult to process the original features i.e, thousands of features. Fuzzy c means clustering algorithm is used for feature extraction for classification.

3) *PCA and Random Projection RP:* Principal Component Analysis (PCA) is a simple technique used to explore and visualize the data easily. PCA extracts useful information from complicated data sets using non parametric method. It determines a lower dimension space by statistical method. Based on eigen value decomposition of the covariance matrix transformation matrix of PCA is calculated and thereby computation cost is more and it is also not suitable for very high dimensional data. The strength of PCA is that there are no parameters to fine tune and also no co-efficient is required to adjust.

Fradkin et al., [51] [52] reported a number of experiments by evaluating random projecton in supervised learning. Different datasets were tested to compare random projection and PCA using several machine learning methods. The results show PCA outperforms RP in efficiency for supervised learning. The results also shows that RP's are well suited to use with nearest neighbour and with SVM classifier and are less satisfactory with decision trees.

Table 2: Summary of The Feature Extraction

Sl.no.	Authors	Models Used for mining	Algorithm	Advantages	Disadvantages
1.	Chen *et al.,* (2016) [45]	-	Temporal pattern miner and Probabilistic Temporal Pattern Miner	Extracts interval based sequential patterns	Closed temporal patterns is not extracted
2.	Bartoli *et al.,* (2016) [46]	Genetic Programming	Regular Expressions	Handles large alphabets effectively	Snippets and faster learning is not considered
3.	Li *et al.,* (2015) [47]	Relevance Feature Discovery 1,2	WFeature Wclustering	Spec function is used for 3 categories and gives best performance,Automatically classifies the terms using Clustering method	Information is not utilized from non relevant documents
4.	Song *et al.,* (2013) [48]	-	Fast clustering based feature selection algorithm	Feature space dimensionality is reduced	Different types of co-relation measure are not explored
5.	Zhong *et al.,* (2012) [44]	Pattern Taxonomy Model	D pattern mining,IP Evolving	PTM is better than pattern based, concept based and term based models	-

III. Pattern Extraction

Patterns which are close to their super patterns that appears in the same paragragh are termed closed relation and needs to be eliminated. The shorter pattern is not considered since it is meaningless while the longer pattern is more meaningful and hence these are significant patterns in the pattern taxonomy.

Sequential Pattern (SP) mining algorithm is used to perform the extraction of frequent sequential pattern. It evaluates pruning and after pruning the final obtained patterns are added to next step of an algorithm. SP mining makes use of projected database method. The advantage of SP mining is that it deals with many sequences simultaneously whereas other techniques can handle one sequence at a time. The interesting information from negative or unlabeled documents are not extracted in this technique.

Abonem et al., [53] presented text mining framework that discovers knowledge by preprocessing the data. Usually text in the documents contains words, special characters and structural information and hence special characters is replaced by symbols. It mainly focuses on refining the uninterested patterns and thus fitering decreases the time and size of search space needed for the discovery phase. It is more efficient when large collection of documents are considered. Post-processing involves pruning, organizing and ordering of the results. The rule of each document is to find a set of characteristics phrases and keywords i.e., length, tightness and mutual confidence. The ranking of the rules within a document is measured by calculating a weight for each rule.

a) *Mining Closed Pattern*

Mining entire set of frequent subsequence for every long pattern generates uncontrollable number of frequent subsequence which are expensive in space and time. Yan et al., [54] proposed a solution for mining only frequent closed subsequence through an algorithm Clospan-Closed Sequential Pattern Mining. Clospan efficiently mines frequent closed sequences in large data sets with low minimum support but does not take advantage of search space pruning property.

Gomariz et al., [55] presented CSpan algorithm for mining closed sequential patterns which mines closed sequential patterns early by using pruning method called occurence checking. CSpan outperforms clospan and claspalgorithm.

Pei et al., [56] proposed CLOSET algorithm for discovering frequent closed itemsets. Three techniques extension of FP growth are developed to reduce the search space and to recognize the frequent closed itemsets. New strategies are constructed based on projection mechanism that makethe mining task scalable and efficient for massive database. CLOSET is faster than earlier methods for finding frequent closed itemsets.

b) *Mining Sequential Pattern*

To delimit the search and to increase the subsequence fragments Han et al., [57] proposed Freespan Frequent Pattern Projected sequential pattern Mining. Freespan fuses the mining of frequent sequence with that of frequent patterns and adopts projected sequence databases. Freespan runs quicker than the Apriori based GSP algorithm. Freespan is highly scalable and processing efficient in mining complete set of patterns. Freespan causes page thrashing as it requires extra memory. With extensive applications in data mining, mining sequential pattern encounters problems with a usage of very large database.

Pei et al., [58] proposed a sequential pattern mining method called Prefix Span(Prefix Projected sequential pattern mining). The complete set of patterns is extracted by reducing the generation of candidate subsequence. Further prefix projection largely reduces projected database size and greatly improves efficiency as shown in Table 3. Making use of RE(Regular Expression) [59] as a flexile constraint SPIRIT algorithm was proposed by Garofalakis et al., [60] for mining patterns that are sequential. A family of four algorithms is executed for forwarding a stronger relaxation of RE. Candidate sequence containing elements are pruned that do not appear in RE than its predecessor in the pattern mining loop.

The degree to which RE constraints are enforced to prune the search space of patterns are the main distinctive factor. The results on the real life data shows RE's adaptability as a user level tool for focussing on interesting patterns.

Jian et al., developed a new framework called Pattern Growth [PG]. PG is based on prefix monotonic property. Every monotonic and anti monotonic regular expression constraints are preprocessed and are pushed into a PG-based mining algorithm. PG adopts and also handles regular expression constraints which is difficult to explore using Apriori based method like SPIRIT. The candidate generation and test framework adopted by PG is less expensive and efficient in pushing many constraints than SPIRIT method. During Prefix growth various irrelevant sequence can be excluded in the huge dataset. Accordingly, projected database quickly shrinks. While PG outperforms SPIRIT, interesting constraints specific to complex structure mining is not be explored.

To filter the discovered patterns, Li et al., [43] [61] proposed an effective pattern discovery technique that deploys and evolves patterns to refine the discovered patterns. Using these discovered patterns, the relevant information can be determined inorder to improve the effectiveness. All frequent short patterns and long patterns are not useful and also long patterns with high specificity suffers from the low problem frequency. The problem of low frequency and misinterpretation for text mining can be solved by employing pattern deploying strategies.

Rather than using individual words, some researches used phrases to discover relevant patterns from documents collection. Hence there is a small improvement in the effectiveness of text mining becauses phrases based methods have consistency of assignment and document frequency for terms to be low. Inje et al., [62] used a pattern based taxonomy(is-a) relation to represent document rather than using single word. The computation cost is reduced by pruning unwanted patterns and hence improves the effectiveness of system.

Bayardo et al., [63] evaluated Max miner algorithm inorder to mine maximal frequent itemsets from large databases. Max- Miner reduces the space of itemsets considered through superset-frequency based pruning. There is a performance improvements over Apriori-like algorithms when frequent itemsets are long and more modest though still substantial improvements when frequent itemsets are short. Completeness at low supports on complex datasets is not achieved.

Jan et al., [64] [65] proposed propositionalization and classification that employs long first order frequent patterns for text mining. The Framework solves three text mining tasks such as information extraction, morphological disambiguation and context sensitive text correction. Propositionalization approach outperforms CBA by using frequent patterns as features. The performance of CBA classifiers greatly depends on number of class association rules and threshold values given by the user. The proposed framework shows that the distributed computation can improve performance of both method since large sample of data and a larger number offeatures are extracted.

Seno et al., [66] proposed an algorithm SLP miner that finds all sequential patterns. It performs effectively satisfying length decreasing support constraint and increases in average length of the sequence. It is expensive as pruning is not considered in this work.

Nizar et al., [67] demonstrates a taxonomy of sequential pattern mining techniques. Reducing the search space can be done by strongly minimizing the support count. Domain knowledge, distributed sequence are not considered in the mining process.

c) *Mining Frequent Sequences*

To extract sequential patterns, various algorithms have been executed by making continuously repeated scans of database and making use of hash structure.

Zaki et al., [68] presented a new novel algorithm SPADE for discovering sequential patterns at a high speed. SPADE decomposes the parent class into small subclasses. These sub problems are executed without depending on other subproblems in main memory by lattice approach. The lattice approach needs only one scan when having some pre-processed data. They also process depth first search and breadth first search for frequent sequence enumeration within each sublattice. By using these search strategies SPADE minimizes the computational costs and I/O costs by reducing number of database scans. It provides pruning strategies to identify the interesting patterns and prune out irrelevant patterns.

BFS outperforms DFS by having more information available for pruning while constructing a set of three sequence, two sequence. BFS require more main memory than DFS. BFS checks the track of idlists for all the classes, while DFS needs to preserve intermediate id lists for two consecutive classes along a specific path.

Han et al., [69] proposed a FP(frequent pattern tree) structure where the complete set of frequent patterns can be extracted by pattern fragment growth. Three techniques are used to achieve mining efficiency compression of the database, (i) FP tree avoids expensive repeatedly scanning database (ii) FP tree prevents generation of large number of candidates sets and uses divide and conquer method which breaks the mining task into a set of tasks that lowers search space. FP growth method [70] is efficient and also scalable for extracting both long and short frequent patterns and it is faster than Apriori algorithm.

Zhang et al., [71] executed CFP Constrained Frequent Pattern algorithm to improve the efficiency of association rule mining. The algorithm is incorporated in an interrelation analysis model for celestial spectra data. The module extracts correlation among the celestial spectra data characteristics. The model do not support for different application domain.

d) Mining Frequent itemsets using Map Reduce

Database Management System have evolved over the last four decades and now functionally rich. Operating and managing very large amount of business data is a challenging task. MapReduce [72] [73] is a framework that process and manages a very large datasets in a distributed clusters efficiently and achieves parallelism.

Xun et al., [74] [75] executed Fidoop algorithm using mapreduce model. Fidoop algorithm uses frequent itemset with different lengths to improve workload balance metric across clusters. Fidoop handles very high dimensional data efficiently but do not work on heterogeneous clusters for mining frequent itemsets.

Wang et al., [76] proposed (FIMMR) Frequent Itemset Mining Mapreduce Framework algorithm. The algorithm initially extracts lower frequent itemset, applies pruning technique and later mines global frequrnt itemset. The speedup of algorithm is satisfactory under low minimum support threshold.

Ramakrishnudu et al., [77] finds infrequent itemset from huge data using mapreduce framework. The efficiency of framework increases as the size of the data is increased. The framework produces few intermediate items during the process.

Ozkural et al., [78] extracts frequent item set by partitioning the graph by a vertex separator. The separator mines the item distribution independently. Parallel frequent itemset algorithm replicates the items that co-relate with the separator. The algorithm minimizes redundancy and load balancing is achieved. Relationship among a very large number of items for real world database is not incorporated.

e) Relevance Feedback Documents

Xu et al., [79] presented a Expectation Maximization(EM) algorithm for relevance feedback inoverlaps in feedback documents. Based on dirichlet compound multinominal(DCM) distribution, EM includes a background collection model reduction, by the methodology of deterministic annealing and query based regularization.

Several Queries which do not contain any relevance feedback needs improvisations by combining pseudo relevance feedback and relevance feedback using a hybrid feed-back paradigm. Instead of using static regularization, the authors adjust the regularization parameter based on the percentage of relevant feedback documents [80]. Further, the design formulates the space for a much newer document progressively. The weighted relevance is computed for an experimental design which further exploits the top retrieved documents by adjusting the selection scheme. The relevance score algorithms need to be validated on several TREC datasets.

Cao et al., [81] re-examined the assumption of most frequent terms in the false feedback documents that are useful and prove that it does not hold in reality. Distinguishing good and bad expansion terms cannot be done in the feedback documents. The difference of term distribution between feedback documents and whole document collection is exploited through the mixture model indicates that good and bad expansion terms may have similar distributions that fails to distinguish. Experiments are conducted to see that each query can keep only the good expansion terms. The new query model integrates the good terms, while classification of term is done to improve the effectiveness of retrieval. In a final query model, the classification score is used to enhance the weight of good terms. Selecting expansion terms are significantly better than traditional expansion terms by evaluating on three TREC datasets. Selection of terms has to be done carefully.

Pak et al., [82] proposed a automatic query expansion algorithm which incorporates a incremental blind approach to choose feedback documents from the top retrieved lists and further finds the terms by aggregating the scores from each feedback document. The algorithm performs significantly better on large documents.

Algarni et al., [83] proposed the adaptive relevance feature discovery(ARFD). Using a sliding window over positive and negative feedback, that ARFD updates the systems knowledge. The system provides a training documents where specific features are discovered. Various methods have been used to merge and revise the weight of the feature in a vector space. Documents are selected based on two categories. The first category is that user provide the interested topic information and the second category is that the user changed the interest topic.

IV. Text Classification and Clustering

Text categorization [84] is a significant issue in text mining. In general, the documents contains large texts and hence it is necessary to classify them into specific classes. Text categorization can be broadly classified into supervised and unsupervised classification. Classifying documents manually is very costly and time consuming task. Hence it is necessary to construct automaic text classifiers using pre-classified sample documents whose time efficiency and accuracy is much better than manual text classification.

Computer programs often treat the document as a sack of words. The main characteristics of text categorization is feature space having high dimensionality. Even for moderate sized text documents, the feature space consists of hundreds and thousands of terms.

Sebastiani et al., [85] reviews the standard approaches that comes under machine learning paradigm for text categorization. The approach also describes the prob- lem faced while document representation constructing classifiers and evaluation of constructed classifier. The experimental study shows comparisons among different classifiers on different versions of reutor dataset. Text categorization is a good benchmark for clarrifying whether a given learning technique can scale up to substantial sizes.

Irfan et al., [86] reviews different pre-processing techniques in text mining to extract various textual patterns from the social networking sites. To explore the unstructured text available from social web, the basic approaches of text mining like classification and clustering are provided.

Wu et al., [87] presents a technique consisting of three preprocessing stages to recognize the text region of huge size and contrast data. A Segmentation algorithm cannot identify the changes that happen both in color and illumination of character in a document image. The technique followsextracting the grayscale image such as from the book cover, magazine RGB plane associated with weighted valve. A multilevel thresholding process is done on each grayscale image independently to identify text region. A recursive filter is executed to interpret which connects components is textual components. An approach to determine score is considered to findout the probabilistic text region of resultant images. If the text region has maximum score, then it is classified as textual component.

V. Discovering Facets for Queries from Search Result

Facets means a phrase or a word. A query facet is a set of items which summarize an important aspect of a query. Dou et al., [88] [89] [90] explores solution of searching the set of facets for a user query. A system called Query Discovery (QD) miner is proposed to mine facets automatically. Expermiments are conducted for 100's of queries and results shows the effectiveness of the system as shown in the table 5. It provides interesting knowledge about a query and however improves searching for the users in different ways. The problem of generating query suggestions based on query facets is not considered that might help users find a better query more easily.

Multifacted search is an important paradigm for extracting and mining applications that provides users to analyze, navigate through multidimensional data.

Facetted search [91] can also be applied on spoken web search problem to index the metadata associated with audio content that provides audio search solution for rural people. The query interface ensures that a user is able to narrow the search results fastly. The approach focuses on indexing system and not generating precision - recall results on a labeled set of data.

Kong et al., [96] incorporated the feedback of users on the query facets into document ranking for evaluating boolean filtering feedback models that are widely used in conventional faceted search which automatically generates the facets for a user given query instead of generating for a complete corpus. The boolean filtering model is less effective than soft ranking models.

Bron et al., [97] proposed a novel framework by adding type filtering based on category information available in wikipedia. Combining a language modelling approach with heuristic based on wikipedia's external links, framework achieves high recall scores by finding homepages of top ranked entities. The model returns entities that have not been judged.

Navarro et al., [98] develops an automatic facet generation framework for an efficient document retrieval. To extract the facets a new approach is developed which is both domain independent and unsupervised. The approach generates multifaceted topic effectively. The subtopics in the text collection is not investigated.

Liu et al., [99] presented the study of exploring topical lead lag across corpora. Selecting which text corpus leads and which lags in a topic is a big challenge. Text pioneer, a visual analytic tool is introduced. The tool investigates lead lag across corpora from global to local level. Multiple perspectives of results are conveyed by two visualizations like global lead lag as hybrid tree, local lead lag as twisted ladder. Text pioneer donot analyze topics within each corpus and across corpora.

Jiang et al., [100] presented Cross Lingual Query Log Topic Model (CL-QLTM) to investigate query logs to derive the latent topics of web search data. The model incorporates different languages by collecting co-occurence relations and cross lingual dictionaries from query log. CL-QLTM is effective and superior in discovering latent topics. The model is not applied on statistical machine translation.

Cafarella et al., [101] exploited the interesting knowledge from webpages which consists of higher relevance to user when compared to traditional approach. The system records co-occurences of schema elements and helps user in navigating, creating synonyms for schema matching use.

Wordnet Domains text document. The queries given by the user is free text queries. Mapping keywords to different attributes and their values of a given entity is a challenging task. Castanet is simple and effective that achieves higher quality results than other automated category creation algorithms. WordNet is not exhaustive and few other mechanism is needed to improve coverage for unknown terms.

Pound et al., [102] proposed a solution that exploits user facetted search behaviour and structured data to find facet utility. The approach captures values and conditional values that provides attributes and values according to user preferences. Experiment results show that the approach is scalable and also outperforms popular commercial systems.

Table 3: Performance of Models to Extract Facets

Sl.no.	Authors	Model	Advantages	Disadvantages
1.	Dou *et al.*, (2016) [88]	Unique Similarity Model	Duplicate List are eliminated	While labelling facets do not identify similar items
2.	Efstathiades *et al.*, (2016) [92]	k Relevant Nearest Neighbour	Relevant point of interest is extracted	Relevance score is not considered
3.	Zhang *et al.*, (2016) [93]	Inverted Linear quadtree	Reduces search space	-
4.	Pripuzie *et al.*, (2015) [94]	Space partition Probing	Top K objects are identified quickly	Fixed bounded region is not incorporated
5.	Hon *et al.*, (2014) [95]	Space Efficient Framework	Robust	Multiple patterns are not handled

Altingovde et al., [103] demonstrate static index pruning technique by incorporating query views like document and term centric. The technique improves the quality of top ranked result. When the web pages changes frequently the original index is not updated.

Koutris et al., [104] proposed a framework for pricing the data based on queries. The polynomial time algorithm is executed for a conjunctive queries of large class and the result shows that the data complexity instance based determincy is CO NP complete. The framework do not explore interaction between pricing and privacy.

Lucchese et al.,[105] proposed two methodlogy for extracting user tasks when they search for relevant data from search engine. The method identifies user query logs and further aggregate same kind of users tasks based on supervised and unsupervised approaches. The method is effective in detecting similar latent needs from a query log. Users task by task search behaviour is not represented in the model.

Liu et al., [106] developed a tool that automatically differentiate structured data from search results. A feature type based approach is introduced which identifies a valid features and evaluates the quality of features using exact and heuristics computation methods. The method achieves local optimality avoids dependency on random initialization. Result differentiation (whether the selected features is interest to users are not) is not incorporated.

Liu et al., [107] proposed matrix representation to discover collection of documents based on user interest. The multidimensional visualization is presented to overcome the difficulty for users to compare across different facet values. The approach further enables visual ordering based on facet values to support cross facet comparisons of items and also support users in exploring tasks. The intradocument details are unavailable and visual scalability is not incorporated.

Efstathiades et al., [92] presents Link of Interest (LOI) to improve the quality of users queries. K Relevant Nearest Neighbor(K-RNN) queries is based on query processing method is proposed to analyse LOI information to retrieve relevant location based point of interest as shown in Table 3. The method captures the relevance aspect of data. Relevance score is not computed.

Hon et al., [95] developed space efficient frame works for top k string retrieval problem that considers two metrics for relevance features which includes frequency and proximity. The threshold based approach on these metrics are also been used. Compact space and sufficient space indexes are derived that results index space and query time with significant robustness. The framework is robust but do not index an the cache oblivious model and also the index takes twice the size of the text. Multiple patterns are not handled.

Zhang et al., [94] proposed (SPP) Space Partition and Probing to keep track of object position and relevance to the query and also to find the vector space. Quality is achieved by using MMR which is one of the important diversification algorithm. The method identifies the next top K object very quickly. SPP helps in reducing object axis and also increases the performance. Fixed bounded region is not considered. Zhang et al., [93] proposed inverted linear quadtree index structure to accomplish both spatial and keyword based techniques to effectively decreases the search space. Spatial keyword queries having two disputes: top k spatial keyword search(TOPK-SK) and batch top k spatial keyword search(BTOPK-SK), in which top-sk fetch the closest k objects which contains all keywords in the query. BTOPK-SK contains set of top k queries. Existing techniques in IL-quadtree presents firstly Keyword first index, which is to extract the related inverted indexes. Partition based method is proposed to further enhance the filtering capabilities of the signature of linear quadtree.

Catallo et al.,[108] proposed probabilistic k- Sky band to process subset of sliding window objects, that are most recent data objects. The algorithm out performs for parameter of large values of K parameter both in memory consumption and time reduction. Adaptive top K processing is not incorporated in the approach.

Bast et al.,[109] presented pre-processing techniques to achieve interactive query times on large text collections. Two similarities measures are considered which includes, firstly, query terms match -similar terms in collection. Secondly, Query terms match -terms with similar prefix in collection which display the results quickly and are more efficient and scalable.

Termehchy et al., [110] introduced the XML stru cture for searching the keyword effectively. Traditional keyword search techniques does not support effectively. In order to overcome these problems for data-centric, XML put forth the Coherency Ranking(CR), which is a database design self sustained ranking method for XML keyword queries that is based on prolonging concepts of data dependencies and mutual information. With the concepts of CR, that analyze the prior approaches to XML keyword search. Approximate coherency ranking and current potent algorithm process queries and rank their responds using CR. CR shows better precision and recall, provides better ranking than prior approaches.

Colini et al., [111] [112] proposes multiple keyword method that provides search auctions with budgets and bidders. Bidders is bounded by multiple slots per keyword. Bidders which have cumulative valuations are click through rates and budgets that confine the overall study of multiple keyword method. Multiple keywords mechanism is compatible, optimal and rational with expectation. In combinatorial setting, each bidder is having a direct involvement in a subset of keywords. Deterministic mechanisms with temper marginal valuations are incompatible.

Wu et al., [113] introduced the concept of safe zones. It studies the moving of top K keyword query. The safe zones saves the time and communication cost. The approach computes safe zone in order to optimize server side computation. It is also used to establish the client server communication. Spatial keyword is not processed and also the safe zone do not compute future path of moving the query.

Lu et al., [114] proposed reverse spatial keyword K nearest neighbour to find the query of object which is similar to one of the neighbour. The query search is based on spatial location and also text associated with it. The algorithm is used to prune unnecessary objects and also computes the lists. The method do not considers textual description of two different objects.

Cao et al., [115] demonstrates the concept of weighing a query. The spatial keywords match considers both the location and the text. The method focuses more on finding queries to group of objects by grouping spatial objects. Top K spatial keyword and weighing of query improves the performance and efficiency. The computational time is reduced but partial coverage of queries is not considered.

VI. Fine Grained Knowledge

Guan et al., [116] suggested “tcpdump” method to capture the web surfing activities from users. Web surfing activities reflects persons fine grained knowledge by recognizing the semantic structures. Further by using Dirichlet process infinite Guassian mixture model is adopted. D-iHMM process is employed for mining the fine grained aspect in each part by session clustering. Discovering fine grained knowledge reflected from people's interaction made knowledge sharing in collaborative environment much easier. Although privacy is major issue.

Wang et al., [117] analysed user's searching behaviors and considered inter-query dependencies. A semi-supervised clustering model is proposed based on the SVM framework. The model enables a more comprehensive understanding of user's searchbehaviors via query search logs and facilitates the development search-engine support for long-term tasks. The performance of the model is superior in identifying cross-session search. User modeling and long-term task based personalization is not considered.

Kotov et al., [118] proposed a method for creating a semi automatically labeled data set that can be used for identifying user's query searches from earlier sessions on the same task and to predict whether a user returns to the same task during his later session. Using logistic regression and MART classifiers the method can effectively model and analyze cross-session of user's information needs. The model is not incorporated in commercial search engines.

VII. Duplicate Detection and Data Fusion

Duplicate detection is the methodology of identification of multiple semantic representation of the existing and similar real world entities. The present day detection methods need to execute larger datasets in the least amount of time and hence to maintain the overall quality of datasets is tougher.

Papenbrock et al., [119] proposed a strategic approach namely the progressive duplicate detection methods as shown in Table 4 which finds the duplicates efficiently and reduces the overall processing time by reporting most of the results as shown in table 7 than the existing classical approaches.

Bano et al., [120] executed innovative windows algorithm that adapts window for duplicates and also which are not duplicates and unnecessary comparisons is avoided.

The duplicate records are a vital problem and a concern in knowledge management [124]. To Extract duplicate data items an entity resolution mechanism is employed for the procedure of cleanup. The overall evaluation reveals that the clustering algorithms perform extraordinarily well with accuracy and f-measure being high.

Whang et al., [125] investigates the enhancement of focusing on several matching records. Three types of hints that are compatible with different ER algorithms:(i) an ordered list of records, (ii) a sorted list of record pairs, (iii) a hierarchy of record partitions. The underlying disadvantage of the process is that it is useful only for database contents.

Duplicate records do not share a strategic key but they build duplicate matching making it a tedious task. Errors are induced because the results of transcription errors, incomplete information and lack of normal formats. Abraham et al., [126] [127] provides survey on different techniques used for detecting duplicates in both XML and relational data. It uses elimination rule to detect duplicates in database.

Elmagarmid et al., [128] present intensive analysis of the literature on duplicate record for detection and covers various similarity metrics, which will detect some duplicate records in exceedingly available information. The strengths of the survey analysis in statistics and machine learning aims to develop a lot of refined matching techniques that deem probabilistic models.

Deduplication is an important issue in the era of huge database [129]. Various indexing techniques have been developed to reduce the number of record pairs to be compared in the matching process. The total candidates generated by these techni- ques have high efficiency with scalability and have been evaluated using various data sets.

The training data in the form of true matches and true non matches is often unavailable in various real-world applications. It is commonly up to domain and linkage experts for decision of the blocking keys. Papadakis et al., [122] presented a blocking methods for clean-clean ER over Highly Heterogeneous Information Spaces (HHIS) through an innovative framework which comprises of two orthogonal layers. The effective layer incorporates methods for construction of several blockings with small probability of hits; the efficiency layer comprises of a rich variety of techniques which restricts the required number of pairwise matches.

Papadakis et al., [123] focuses to boost the overallblocking efficiency of the quadratic task on Entity Resolution among large, noisy, and heterogeneous information areas.

The problem of merging many large databases is often encountered in KDD. It is usually referred to as the Merge/Purge problem and is difficult to resolve in scale and accuracy. The Record linkage [130] is a well-known data integration strategy that uses sets for merging, matching and elimination of duplicate records in large and heterogeneous databases. The suffix grouping methodology facilitates the causal ordering used by the indexes

Table 4: Algorithms for Duplicate Detection

Sl.no.	Authors	Algorithm	Window selection	Advantages	Disadvantages
1.	Papenbrock *et al.,* (2015) [119]	PSNM	Adaptive	Efficient with limited execution time	Delivers results moderately
2.	Bano *et al.,* (2015) [120]	Innovative Window	Adaptive	Unnecessary Comparison is avoided	Do not support on Multiple Datasets
3.	Bronselaer *et al.,* (2015) [121]	Fusion Propogation	-	Conflicts in relationship attributes are resolved	More Expensive
4.	Papadakis *et al.,* (2013) [122]	Attribute Clustering	-	Effective on real world datasets	low quality blocks, Parallelizing is not adopted
5.	Papadakis *et al.,* (2011) [123]	-	Adaptive	Time Complexity is reduced	Process is very slow

for merging blocks with least marginal extra cost resulting in high accuracy. An efficient grouping similar suffixes is carried out with incorporation of a sliding window technique. The method is helpful in various health records for understanding patient's details but is not very efficient as it concentrates only on blocking and not on windowing technique. Additionally the methodology with duplicates that are detected using the state of the art expandable paradigm is approximate [131]. It is quite helpful in creating clusters records.

Bronselaer et al., [132] focused on Information aggregation approach which combine information and rules available from independent sources into summarization. Information aggregation is investigated in the context of inferencing objects from several entity relations. The complex objects are composed of merge functions for atomic and subatomic objects in a way that the composite function inherits the properties of the merge functions.

Sorted Neighborhood Method (SNM) proposed by Draisbach et al., [133] partitions data set and comparison are performed on the jurisdiction of each partition. Further, the advances in a window over the data is done by comparison of the records that appears within the range of same window. Duplicate Count Strategy (DCS) which is a variation of SNM is proposed by regulating the window size. DCS++ is proposed which is much better than the original SNM in terms of efficiency but the disadvantage is that the window size is fixed and is expensive for selection and operation. Some duplicates might be missed when large window are used.

The tuples in the relational structure of the database give an overview of the similar real world entities such tuples are described as duplicates. Deleting these duplicates and in turn facilitating their replacement with several other tuples represents the joint informational structure of the duplicate tuples up to a maximum level. The incorporated delete and then replacement mode of operation is termed as fusion. The removal of the original duplicate tuples can deviate from the referential integrity.

Bronselaer et al., [121] describes a technique to maintain the referential integrity. The fusion Propogation algorithm is based on first and second order fusion derivatives to resolve conflicts and clashes. Traditional referential integrity strategies like DELETE cascading, are highly sophisticated. Execution time and recursively calling the propagation algorithm increases when the length of chain linked relations increases.

Bleiholder et al., proposes the SQL Fuse by inducing the schema and semantics. The existential approach is towards the architecture, query languages, and query execution. The final step of actually aggregating data from multiple heterogeneous sources into a consistent and homogeneous datasetand is often inconsiderable.

Naumann et al., [134] observes that amount of noisy data are in abundance from several data sources. Without any suitable techniques for integrating and fusing noisy data with deviations, the quality of data associated with an integrated system remains extremely low. It is necessary for allowing tentative and declarative integration of noisy and scattered data by incorporating schema matching, duplicate detection and fusion. Subjected to SQL-like query against a series of tables instance, oriented schema matching covers the cognitive bridge of the varied tables by alignment of various corresponding attributes. Further, a duplicate detection technique is used for multiple representations of several matching entities. Finally, the paradigm of data fusion for resolving a conflict in turn merges around with each individualistic duplicate transforming it into a unique singular representation.

Bleiholder et al., [135] explains a conceptual understanding of classification of different operators over data fusion. Numerous techniques are based on standard and advanced operators of algebraic relations and SQL. The concept of Co-clustering is explained from several techniques for tapping the rich and associated meta tag information of various multimedia web documents that includes annotations, descriptions and associations. Varied Coclustering mechanisms are proposed for linked data that are obtained from multiple sources which do not matter the representational problem of precise texts but rather increase their performance up to the most minimally empirical measurement of the multi-modal features.

The two channel Heterogeneous Fusion ART (HF-ART) yields several multiple channels divergently. The GHF-ART [136] is designed to effectively represent multimedia content that incorporates Meta data to handle precise and noisy texts. It is not trained directly using the text features but can be identified as a key tag by training it with the probabilistic distribution of the tag based occurrences. The approach also incorporates a highly and the most adaptive methodology for active and efficient fusion of multimodal.

VIII. Conclusions

The paper presents different techniques and framework to extract relevant features from huge amount of unstructured text documents. The paper also reviews a survey on various text classification, clustering, summerization methods.

To guarantee the quality of extracted relevant features in a collection of text documents is a great challenge. Many text mining techniques have been proposed till date. However how effectively the discovered features is interesting and useful to the user is an open issue.

Our future work is to efficiently utilize relevant documents from non relevant documents. Effective filtering model is required to automatically generate facets. The security and time to extract the useful features that is duplicate free and fine grained knowledge helps the user to reduce time in searching various web pages needs to be addressed.

Références

1. R. Agrawal and M. Batra, “A Detailed Study on Text Mining Techniques,” International Journal of Soft Computing and Engineering (IJSCE) ISSN, vol. 2, no. 6, pp. 2231–2307, 2013.
2. V. H. Bhat, P. G. Rao, R. Abhilash, P. D. Shenoy, K. R. Venugopal, and L. Patnaik, “A Data Mining Approach for Data Generation and Analysis for Digital Forensic Application,” International Journal of Engineering and Technology, vol. 2, no. 3, pp. 313–319, 2010.
3. Y. Zhang, M. Chen, and L. Liu, “A Review on Text Mining,” In Proceedings of 6th IEEE International Conference on Software Engineering and Service Science (ICSESS), pp. 681– 685, 2015.
4. S. Shehata, F. Karray, and M. S. Kamel, “An Efficient Concept-based Mining Model for Enhancing Text Clustering,” IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1360–1371, 2010.
5. V. H. Bhat, P. G. Rao, R. Abhilash, P. D. Shenoy, K. R. Venugopal, and L. Patnaik, “A Novel Data Generation Approach for Digital Forensic Application in Data Mining,” In Proceedings of Second International Conference on Machine Learning and Computing (ICMLC), pp. 86–90, 2010.
6. D. E. Brown, “Text Mining the Contributors to Rail Accidents,” IEEE Transactions on Intelligent Transportation Systems, vol. 27, no. 5, pp. 1–10, 2015.
7. K. R. Venugopal, K. Srinivasa, and L. M. Patnaik, “Soft Computing for Data Mining Applications,” Springer, 2009.
8. V. K. Verma, M. Ranjan, and P. Mishra, “Text Mining and Information Professionals: Role, Issues and Challenges,” In Proceedings of 4th International Symposium on Emerging Trends and Technologies in Libraries and Information Services (ETTLIS), pp. 133–137, 2015.
9. A. Akilan, “Text mining: Challenges and Future Directions,” In Proceedings of Second International Conference on Electronics and Communication Systems (ICECS), pp. 1679–1684, 2015.
10. D. Sanchez, M. J. Martin-Bautista, I. Blanco, and C. Torre, “Text Knowledge Mining:An Alternative to Text Data Mining,” In Proceedings of IEEE International Conference on Data Mining Workshops(ICDMW), pp. 664–672, 2008.
11. Y. Dai, T. Kakkonen, and E. Sutinen, “Minedec:A Decision- Support Model that Combines Text-mining Technologies with Two Competitive Intelligence Analysis Methods,” International Journal of Computer Information Systems and Industrial Management Applications, vol. 3, no. 10, pp. 165–173,2011.
12. Y. Hu and X. Wan, “Ppsgen: Learning-Based Presentation Slides Generation for Academic Papers,” IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 4, pp. 1085– 1097, 2015.
13. E. D’Andrea, P. Ducange, B. Lazzerini, and F. Marcelloni, “Real-Time Detection of Traffic from Twitter Stream Analysis,” IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 4, pp. 2269–2283, 2015.
14. R. Li, K. H. Lei, R. Khadiwala, and K. C.-C. Chang, “Tedas: A Twitter-based Event Detection and

Analysis System," In Proceedings of IEEE 28th International Conference on Data Engineering (ICDE), pp. 1273–1276, 2012.

15. R. Parikh and K. Karlapalem, "Et: Events from Tweets," In Proceedings of the 22nd International Conference on World Wide Web Companion, pp. 613–620, 2013.
16. V. H. Bhat, V. R. Malkani, P. D. Shenoy, K. R. Venugopal, and L. Patnaik, "Classification of Email using Beaks: Behavior and Keyword Stemming," In Proceedings of IEEE Region 10 Conference TENCON, pp. 1139–1143, 2011.
17. A. Schulz, P. Ristoski, and H. Paulheim, "I See a Car Crash: Real-Time Detection of Small Scale Incidents in Microblogs," The Semantic Web: ESWC Satellite Events, pp. 22–33, 2013.
18. A. Gonzalez, L. M. Bergasa, and J. J. Yebes, "Text Detection and Recognition on Traffic Panels from Street-Level Imagery using Visual Appearance," IEEE Transactions on Intelligent Transportation Systems, vol. 15, no. 1, pp. 228–238, 2014.
19. D. P. Muni, N. R. Pal, and J. Das, "Genetic Programming for Simultaneous Feature Selection and Classifier Design," IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 36, no. 1, pp. 106–117, 2006.
20. M. H. Aghdam, N. Ghasem-Aghaee, and M. E. Basiri, "Text Feature Selection Using Ant Colony Optimization," Expert Systems with Applications, vol. 36, no. 3, pp. 6843–6853, 2009.
21. K. Srinivasa, A. Singh, A. Thomas, K. R. Venugopal, and L. Patnaik, "Generic Feature Extraction for Classification using Fuzzy C-means Clustering," In Proceedings of 3rd International Conference on Intelligent Sensing and Information Processing, pp. 33–38, 2005.
22. E. Gasca, J. S. S´anchez, and R. Alonso, "Eliminating Redundancy and Irrelevance using a New Mlp-based Feature Selection Method," Pattern Recognition, vol. 39, no. 2, pp. 313–315, 2006.
23. R. K. Sivagaminathan and S. Ramakrishnan, "A Hybrid Approach for Feature Subset Selection using Neural Networks and Ant Colony Optimization," Expert systems with Applications, vol. 33, no. 1, pp. 49–60, 2007.
24. D. Cai, C. Zhang, and X. He, "Unsupervised Feature Selection for Multi-cluster Data," In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 333–342, 2010.
25. Z. Zhao, X. He, D. Cai, L. Zhang, W. Ng, and Y. Zhuang, "Graph Regularized Feature Selection with Data Reconstruction," IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 3, pp. 689–700, 2016.
26. L. Xu, C. Jiang, Y. Ren, and H.-H. Chen, "Microblog Dimensionality Reduction—A Deep Learning Approach," IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 7, pp. 1779–1789, 2016.
27. D. Wang, F. Nie, and H. Huang, "Feature Selection via Global Redundancy Minimization," IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 10, pp. 2743–2755, 2015.
28. H. Ogura, H. Amano, and M. Kondo, "Feature Selection with a Measure of Deviations from Poisson in Text Categorization," Expert Systems with Applications, vol. 36, no. 3, pp. 6826 – 6832, 2009.
29. N. Azam and J. Yao, "Comparison of Term Frequency and Document Frequency based Feature Selection Metrics in Text Categorization," Expert Systems with Applications, vol. 39, no. 5, pp. 4760–4768, 2012.
30. S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph Embedding and Extensions: A General Framework for Dimensionality Reduction," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 1, pp. 40–51, 2007.
31. Z. Zhao, L. Wang, H. Liu, and J. Ye, "On Similarity Preserving Feature Selection," IEEE Transactions on Knowledge and Data Engineering, vol. 25, no. 3, pp. 619–632, 2013.
32. Z. Zhao, X. He, L. Zhang, W. Ng, and Y. Zhuang, "Graph Regularized Feature Selection with Data Reconstruction," IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 3, pp. 289–700, 2016.
33. J. Tang and H. Liu, "Unsupervised Feature Selection for Linked Social Media Data," In Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 904–912, 2012.
34. H. Mousavi, D. Kerr, M. Iseli, and C. Zaniolo, "Harvesting Domain Specific Ontologies from Text," In Proceedings of IEEE International Conference on Semantic Computing (ICSC), pp. 211–218, 2014.
35. Y. Yang, H. T. Shen, Z. Ma, Z. Huang, and X. Zhou, "l2, 1- norm Regularized Discriminative Feature Selection for Unsupervised Learning," In Proceedings of the International Joint Conference on Artificial Intelligence(IJCAI), vol. 22, no. 1, pp. 1589–1594, 2011.
36. D. Cai, X. He, J. Han, and T. S. Huang, "Graph Regularized Nonnegative Matrix Factorization for Data Representation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, no. 8, pp. 1548–1560, 2011.
37. D. Sejal, K. Shailesh, V. Tejaswi, D. Anvekar, K. R. Venugopal, S. Iyengar, and L. Patnaik, "Qrgqr: Query Relevance Graph for Query Recommendation," In Proceedings of IEEE Region 10 Symposium (TENSYMP), pp. 78–81, 2015.
38. W. Fan, N. Bouguila, and D. Ziou, "Unsupervised Hybrid Feature Extraction Selection for High-

Dimensional Non-Gaussian Data Clustering with Variational Inference," IEEE Transactions on Knowledge and Data Engineering, vol. 25, no. 7, pp. 1670–1685, 2013.

39. V. Gupta and G. S. Lehal, "A Survey of Text Summarization Extractive Techniques," Journal of Emerging Technologies in Web Intelligence, vol. 2, no. 3, pp. 258–268, 2010.
40. P. S. Negi, M. Rauthan, and H. Dhami, "Text Summarization for Information Retrieval using Pattern Recognition Techniques," International Journal of Computer Applications, vol. 21, no. 10, pp. 20–24, 2011.
41. F. Debole and F. Sebastiani, "An Analysis of the Relative Hardness of Reuters-21578 Subsets," Journal of the American Society for Information Science and Technology, vol. 56, no. 6, pp. 584–596, 2005.
42. F. Xie, X. Wu, and X. Hu, "Keyphrase Extraction based on Semantic Relatedness," In Proceedings of 9th IEEE International Conference on Cognitive Informatics (ICCI), pp. 308– 312, 2010.
43. Y. Li, A. Algarni, and N. Zhong, "Mining Positive andNegative Patterns for Relevance Feature Discovery," In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 753–762, 2010.
44. N. Zhong, Y. Li, and S.-T. Wu, "Effective Pattern Discovery for Text Mining," IEEE Transactions on Knowledge and Data Engineering, vol. 24, no. 1, pp. 30–44, 2012.
45. Y.-C. Chen, W.-C. Peng, and S.-Y. Lee, "Mining Temporal Patterns in Time Interval-Based Data," IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 12, pp. 3318– 3331, 2015.
46. A. Bartoli, A. De Lorenzo, E. Medvet, and F. Tarlao, "Inference of Regular Expressions for Text Extraction from Examples," IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 5, pp. 1217–1230, 2016.
47. Y. Li, A. Algarni, M. Albathan, Y. Shen, and M. A. Bijaksana, "Relevance Feature Discovery for Text Mining," IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 6, pp. 1656–1669, 2015.
48. Q. Song, J. Ni, and G. Wang, "A Fast Clustering-based Feature Subset Selection Algorithm for High-Dimensional Data," IEEE Transactions on Knowledge and Data Engineering, vol. 25, no. 1, pp. 1–14, 2013.
49. T.-S. Nguyen, H. W. Lauw, and P. Tsaparas, "Review Selection Using Micro-Reviews," IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 4, pp. 1098–1111, 2015.
50. A. I. Kadhim, Y. Cheah, N. H. Ahamed, L. A. Salman et al., "Feature Extraction for Co-occurrence-based Cosine Similarity Score of Text Documents," In Proceedings of IEEE Student Conference on Research and Development (SCOReD), pp. 1–4, 2014.
51. D. Fradkin and D. Madigan, "Experiments with Random Projections for Machine Learning," In Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 517–522, 2003.
52. S. Joshi, D. Shenoy, P. rashmi, K. R. Venugopal, and L. Patnaik, "Classification of Alzheimer's Disease and Parkinson's Disease by using Machine Learning and Neural Network Methods," In Proceedings of Second International Conference on Machine Learning and Computing (ICMLC), pp. 218–222, 2010.
53. H. Ahonen, O. Heinonen, M. Klemettinen, and A. I. Verkamo, "Applying Data Mining Techniques for Descriptive Phrase Extraction in Digital Document Collections," In Proceedings of IEEE International Forum on Research and Technology Advances in Digital Libraries, pp. 2–11, 1998.
54. X. Yan, J. Han, and R. Afshar, "Clospan: Mining Closed Sequential Patterns in Large Datasets," In SDM, pp. 166–177, 2003.
55. A. Gomariz, M. Campos, R. Marin, and B. Goethals, "Clasp: An Efficient Algorithm for Mining Frequent Closed Sequences," Advances in Knowledge Discovery and Data Mining, pp. 50–61, 2013.
56. J. Pei, J. Han, R. Mao et al., "Closet: An Efficient Algorithm for Mining Frequent Closed Itemsets," ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, vol. 4, no. 2, pp. 21–30, 2000.
57. J. Han, J. Pei, B. Mortazavi-Asl, Q. Chen, U. Dayal, and M.-C. Hsu, "Freespan: Frequent Pattern-Projected Sequential Pattern Mining," In Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 355–359, 2000.
58. J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M.-C. Hsu, "Prefixspan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth," ICCN, pp. 215– 224, 2001.
59. K. R. Venugopal and R. Buyya, "Mastering c++," Tata McGraw-Hill Education, 2013.
60. M. N. Garofalakis, R. Rastogi, and K. Shim, "Spirit: Sequential Pattern Mining with Regular Expression Constraints," VLDB, vol. 99, pp. 7–10, 1999.
61. N. Zhong, Y. Li, and S.-T. Wu, "Effective Pattern Discovery for Text Mining," IEEE Transactions on Knowledge and Data Engineering, vol. 24, no. 1, pp. 30–44, 2012.
62. A. Inje and U. Patil, "Operational Pattern Revealing Technique in Text Mining," In Proceedings of IEEE Students' Conference on Electrical, Electronics and Computer Science (SCEECS), pp. 1–5, 2014.
63. R. J. Bayardo Jr, "Efficiently Mining Long Patterns from Databases," ACM Sigmod Record, vol. 27, no. 2, pp. 85–93, 1998.

64. Y. Yang and J. O. Pedersen, “A Comparative Study on Feature Selection in Text Categorization,” ICML, vol. 97, pp. 412–420, 1997.
65. P. D. Shenoy, K. Srinivasa, and L. M. K R Venugopal, Patnaik, “Dynamic Association Rule Mining using Genetic Algorithms,” Intelligent Data Analysis, vol. 9, no. 5, pp. 439– 453, 2005.
66. M. Seno and G. Karypis, “Slpminer: An Algorithm for Finding Frequent Sequential Patterns using Length-Decreasing Support Constraint,” In Proceedings of IEEE International Conference on Data Mining, pp. 418–425, 2002.
67. N. R. Mabroukeh and C. I. Ezeife, “A Taxonomy of Sequential Pattern Mining Algorithms,” ACM Computing Surveys (CSUR), vol. 43, no. 1, pp. 1–41, 2010.
68. M. J. Zaki, “Spade: An Efficient Algorithm for Mining Frequent Sequences,” Machine learning, vol. 42, no. 1-2, pp. 31– 60, 2001.
69. J. Han, J. Pei, Y. Yin, and R. Mao, “Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach,” Data Mining and Knowledge Discovery, vol. 8, no. 1, pp. 53–87, 2004.
70. V. P. Raju and G. S. Varma, “Mining Closed Sequential Patterns in Large Sequence Databases,” International Journal of Database Management Systems, vol. 7, no. 1, pp. 29–40, 2015.
71. J. Zhang, X. Zhao, S. Zhang, S. Yin, X. Qin, and I. Senior Member, “Interrelation Analysis of Celestial Spectra Data using Constrained Frequent Pattern Trees,” Knowledge-Based Systems, vol. 41, no. 4, pp. 77–88, 2013.
72. F. Li, B. C. Ooi, M. T. O¨ zsu, and S. Wu, “Distributed Data Management using Mapreduce,” ACM Computing Surveys (CSUR), vol. 46, no. 3, pp. 31–42, 2014.
73. N. Tiwari, S. Sarkar, U. Bellur, and M. Indrawan, “Classification Framework of Mapreduce Scheduling Algorithms,” ACM Computing Surveys (CSUR), vol. 47, no. 3, pp. 49–88, 2015.
74. Y. Xun, J. Zhang, and X. Qin, “Fidoop: Parallel Mining of Frequent Itemsets Using Mapreduce,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 46, no. 3, pp. 313–325, 2016.
75. S. Sakr, A. Liu, and A. G. Fayoumi, “The Family of Mapreduce and Large-Scale Data Processing Systems,” ACM Computing Surveys (CSUR), vol. 46, no. 1, pp. 11–44, 2013.
76. L. Wang, L. Feng, J. Zhang, and P. Liao, “An Efficient Algorithm of Frequent Itemsets Mining based on Mapreduce,” Journal of Information and Computational Science, vol. 11, no. 8, pp. 2809–2816, 2014.
77. T. Ramakrishnudu and R. Subramanyam, “Mining Interesting Infrequent Itemsets from Very Large Data based on Mapreduce Framework,” International Journal of Intelligent Systems and Applications, vol. 7, no. 7, pp. 44–64, 2015.
78. E. Ozkural, B. Ucar, and C. Aykanat, “Parallel Frequent Item Set Mining with Selective Item Replication,” IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 10, pp. 1632–1640, 2011.
79. Z. Xu and R. Akella, “Active Relevance Feedback for Difficult Queries,” In Proceedings of the 17th ACM Conference on Information and Knowledge Management, pp. 459–468, 2008.
80. S. Desai, V. Chandrasheker, V. Mathapati, K. R. V. Rajuk, S. S. Iyengar, and L. M. Patnaik, “User Feedback Session with Clicked and Unclicked Documents for Related Search Recommendation,” IADIS-International Journal on Computer Science and Information Systems, vol. 11, no. 1, pp. 81–98, 2016.
81. G. Cao, J.-Y. Nie, J. Gao, and S. Robertson, “Selecting Good Expansion Terms for Pseudo-Relevance Feedback,” In Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 243–250, 2008.
82. J. H. Paik, D. Pal, and S. K. Parui, “Incremental Blind Feedback: An Effective Approach to Automatic Query Expansion,” ACM Transactions on Asian Language Information Processing (TALIP), vol. 13, no. 3, pp. 13–35, 2014.
83. A. Algarni, Y. Li, and Y. Xu, “Selected New Training Documents to Update User profile,” In Proceedings of the 19th ACM International Conference on Information and Knowledge Management, pp. 799–808, 2010.
84. S. Niharika, V. S. Latha, and D. Lavanya, “A Survey on Text Categorization,” International Journal of Computer Trends and Technology, vol. 3, no. 1, pp. 39–45, 2012.
85. F. Sebastiani, “Machine Learning in Automated Text Categorization,” ACM Computing Surveys (CSUR), vol. 34, no. 1, pp. 1–47, 2002.
86. R. Irfan, C. K. King, D. Grages, S. Ewen, S. U. Khan, S. A. Madani, J. Kolodziej, L. Wang, D. Chen, A. Rayes et al., “A Survey on Text Mining in Social Networks,” The Knowledge Engineering Review, vol. 30, no. 2, pp. 157–170, 2015.
87. H. N. Vu, T. A. Tran, I. S. Na, and S. H. Kim, “Automatic Extraction of Text Regions from Document Images by Multilevel Thresholding and k-means Clustering,” In Proceedings of IEEE/ACIS 14th International Conference on Computer and Information Science (ICIS), pp. 329–334, 2015.
88. Z. Dou, Z. Jiang, S. Hu, J.-R. Wen, and R. Song, “Automatically Mining Facets for Queries from Their Search Results,” IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 2, pp. 385–397, 2016.

89. A. Sejal, K. Shailesh, V. Tejaswi, D. Anvekar, K. R. Venugopal, S. Iyengar, and L. Patnaik, “Query Click and Text Similarity Graph for Query Suggestions,” International Workshop on Machine Learning and Data Mining in Pattern Recognition, pp. 328–341, 2015.
90. X. Shi and C. C. Yang, “Mining Related Queries from Web Search Engine Query Logs using an Improved Association Rule Mining Model,” Journal of the American Society for Information Science and Technology, vol. 58, no. 12, pp. 1871–1883, 2007.
91. M. Diao, S. Mukherjea, N. Rajput, and K. Srivastava, “Faceted Search and Browsing of Audio Content on Spoken Web,” In Proceedings of the 19th ACM International Conference on Information and Knowledge Management, pp. 1029–1038, 2010.
92. C. Efstathiades, A. Efentakis, and D. Pfoser, “Efficient Processing of Relevant Nearest-Neighbor Queries,” ACM Transactions on Spatial Algorithms and Systems (TSAS), vol. 2, no. 3, pp. 9–37, 2016.
93. C. Zhang, Y. Zhang, W. Zhang, and X. Lin, “Inverted Linear Quadtree: Efficient Top k Spatial Keyword S219219earch,” IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 7, pp. 1706–1721, 2016.
94. K. Pripuˇzi´c, I. P. ˇ Zarko, and K. Aberer, “Time - and Space-Efficient Sliding Window Top-k Query Processi- ng,” ACM Transactions on Database Systems (TODS), vol. 40, no. 1, pp. 1–36, 2015.
95. W.-K. Hon, R. Shah, S. V. Thankachan, and J. S. Vitter, “Space-Efficient Frameworks for Top-k String Retrieval,” Journal of the ACM (JACM), vol. 61, no. 2, pp. 9–45, 2014.
96. W. Kong and J. Allan, “Extending Faceted Search to the General Web,” In Proceedings of the 23rd ACM International Conference on Information and Knowledge Management, pp. 839–848, 2014.
97. M. Bron, K. Balog, and M. De Rijke, “Ranking Related Entities: Components and Analyses,” In Proceedings of the 19th ACM International Conference on Information and Knowledge Management, pp. 1079–1088, 2010.
98. G. Navarro, “Spaces, Trees, and Colors: The Algorithmic Landscape of Document Retrieval on Sequences,” ACM Computing Surveys (CSUR), vol. 46, no. 4, pp. 52–99, 2014.
99. S. Liu, Y. Chen, H. Wei, J. Yang, K. Zhou, and S. M. Drucker, “Exploring Topical Lead-Lag Across Corpora,” IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 1, pp. 115–129, 2015.
100. D. Jiang, Y. Tong, and Y. Song, “Cross-Lingual Topic Discovery from Multilingual Search Engine Query Log,” ACM Transactions on Information Systems (TOIS), vol. 35, no. 2, pp. 9–37, 2016.
101. M. J. Cafarella, A. Halevy, D. Z. Wang, E. Wu, and Y. Zhang, “Webtables: Exploring the Power of Tables on the Web,” In Proceedings of the VLDB Data to find Endowment, vol. 1, no. 1, pp. 538–549, 2008.
102. J. Pound, S. Paparizos, and P. Tsaparas, “Facet Discovery for Structured Web Search: A Query-Log Mining Approach,” In Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 169–180, 2011.
103. I. S. Altingovde, R. Ozcan, and O¨ . Ulusoy, “Static Index Pruning in Web Search Engines: Combining Term and Document Popularities with Query Views,” ACM Transactions on Information Systems (TOIS), vol. 30, no. 1, pp. 2–30, 2012.
104. P. Koutris, P. Upadhyaya, M. Balazinska, B. Howe, and D. Suciu, “Query-Based Data Pricing,” Journal of the ACM (JACM), vol. 62, no. 5, pp. 43–87, 2015.
105. C. Lucchese, S. Orlando, R. Perego, F. Silvestri, and G. Tolomei, “Discovering Tasks from Search Engine Query Logs,” ACM Transactions on Information Systems (TOIS), vol. 31, no. 3, pp. 14–58, 2013.
106. Z. Liu and Y. Chen, “Differentiating Search Results on Structured Data,” ACM Transactions on Database Systems (TODS), vol. 37, no. 1, pp. 4–34, 2012.
107. V. Thai, P.-Y. Rouille, and S. Handschuh, “Visual Abstraction and Ordering in Faceted Browsing of Text Collections,” ACM Transactions on Intelligent Systems and Technology (TIST), vol. 3, no. 2, pp. 21–45, 2012.
108. Catallo, E. Ciceri, P. Fraternali, D. Martinenghi, and M. Tagliasacchi, “Top-k Diversity Queries Over Bounded Regions,” ACM Transactions on Database Systems (TODS), vol. 38, no. 2, pp. 10–55, 2013.
109. H. Bast and M. Celikik, “Efficient Fuzzy Search in Large Text Collections,” ACM Transactions on Information Systems (TOIS), vol. 31, no. 2, pp. 10–69, 2013.
110. A. Termehchy and M. Winslett, “Using Structural Information in Xml Keyword Search Effectively,” ACM Transactions on Database Systems (TODS), vol. 36, no. 1, pp. 4–44, 2011.
111. R. Colini-Baldeschi, S. Leonardi, M. Henzinger, and M. Starnberger, “On Multiple Keyword Sponsored Search Auctions with Budgets,” ACM Transactions on Economics and Computation, vol. 4, no. 1, pp. 2–36, 2016.
112. Arguello and R. Capra, “The Effects of Aggregated Search Coherence on Search Behavior,” ACM Transactions on Information Systems (TOIS), vol. 35, no. 1, pp. 2–32, 2016.
113. D. Wu, M. L. Yiu, and C. S. Jensen, “Moving Spatial Keyword Queries: Formulation, Methods, and Analysis,” ACM Transactions on Database Systems (TODS), vol. 38, no. 1, pp. 7–55, 2013.
114. Y. Lu, J. Lu, G. Cong, W. Wu, and C. Shahabi, “Efficient Algorithms and Cost Models for Reverse Spatial-Keyword K Nearest Neighbor Search,” ACM

Transactions on Database Systems (TODS), vol. 39, no. 2, pp. 13–61, 2014.

115. X. Cao, G. Cong, T. Guo, C. S. Jensen, and B. C. Ooi, "Efficient Processing of Spatial Group Keyword Queries," ACM Transactions on Database Systems (TODS), vol. 40, no. 2, pp. 13–59, 2015.

116. Z. Guan, S. Yang, H. Sun, M. Srivatsa, and X. Yan, "Fine- Grained Knowledge Sharing in Collaborative Environments," IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 8, pp. 2163–2174, 2015.

117. H. Wang, Y. Song, M.-W. Chang, X. He, R. W. White, and W. Chu, "Learning to Extract Cross-Session Search Tasks," In Proceedings of the 22nd International Conference on World Wide Web, pp. 1353–1364, 2013.

118. A. Kotov, P. N. Bennett, R. W. White, S. T. Dumais, and J. Teevan, "Modeling and Analysis of Cross-Session Search Tasks," In Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 5–14, 2011.

119. T. Papenbrock, A. Heise, and F. Naumann, "Progressive Duplicate Detection," IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 5, pp. 1316–1329, 2015.

120. H. Bano and F. Azam, "Innovative Windows for Duplicate Detection," International Journal of Software Engineering and Its Applications, vol. 9, no. 1, pp. 95–104, 2015.

121. A. Bronselaer, D. Van Britsom, and G. De Tre, "Propagation of Data Fusion," IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 5, pp. 1330–1342, 2015.

122. G. Papadakis, E. Ioannou, T. Palpanas, C. Nieder´ee, and W. Nejdl, "A Blocking Framework for Entity Resolution in Highly Heterogeneous Information Spaces," IEEE Transactions on Knowledge and Data Engineering, vol. 25, no. 12, pp. 2665–2682, 2013.

123. G. Papadakis and W. Nejdl, "Efficient Entity Resolution Methods for Heterogeneous Information Spaces," In Proceedings of IEEE 27th International Conference on Data Engineering Workshops (ICDEW), pp. 304–307, 2011.

124. O. Hassanzadeh, F. Chiang, H. C. Lee, and R. J. Miller, "Framework for Evaluating Clustering Algorithms in Duplicate Detection," In Proceedings of the VLDB Endowment, vol. 2, no. 1, pp. 1282–1293, 2009.

125. S. E. Whang, D. Marmaros, and H. Garcia-Molina, "Pay-asyou- go Entity Resolution," IEEE Transactions on Knowledge and Data Engineering, vol. 25, no. 5, pp. 1111–1124, 2013.

126. A. A. Abraham and S. D. Kanmani, "A Survey on Various Methods used for Detecting Duplicates in Xml Data," International Journal of Engineering Research and Technology, vol. 3, no. 1, pp. 1–10, 2014.

127. J. J. Tamilselvi and C. B. Gifta, "Handling Duplicate Data in Data Warehouse for Data Mining," International Journal of Computer Applications (0975–8887), vol. 15, no. 4, pp. 1–9, 2011.

128. A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, "Duplicate Record Detection: A Survey," IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 1, pp. 1–16, 2007.

129. P. Christen, "A Survey of Indexing Techniques for Scalable Record Linkage and Deduplication," IEEE Transactions on Knowledge and Data Engineering, vol. 24, no. 9, pp. 1537– 1555, 2012.

130. T. De Vries, H. Ke, S. Chawla, and P. Christen, "Robust Record Linkage Blocking using Suffix Arrays and Bloom Filters," ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 5, no. 2, pp. 9–44, 2011.

131. O. Hassanzadeh and R. J. Miller, "Creating Probabilistic Databases from Duplicated Data," The VLDB Journal—The International Journal on Very Large Data Bases, vol. 18, no. 5, pp. 1141–1166, 2009.

132. A. Bronselaer and G. De Tr´e, "Aspects of object Merging," Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS), pp. 1–6, 2010.

133. U. Draisbach, F. Naumann, S. Szott, and O. Wonneberg, "Adaptive Windows for Duplicate Detection," In Proceedings of IEEE 28th International Conference on Data Engineering (ICDE), pp. 1073–1083, 2012.

134. F. Naumann, A. Bilke, J. Bleiholder, and M. Weis, "Data Fusion in Three Steps: Resolving Schema, Tuple, and Value Inconsistencies." IEEE Data Engineering and Management, vol. 29, no. 2, pp. 21–31, 2006.

135. J. Bleiholder and F. Naumann, "Data Fusion," ACM Computing Surveys (CSUR), vol. 41, no. 1, pp. 1–41, 2009.

136. L. Meng, A.-H. Tan, and D. Xu, "Semi-Supervised Heterogeneous Fusion for Multimedia Data Co-Clustering," IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 9, pp. 2293–2306, 2014.