In the Name of God

Journal of
Information Systems & Telecommunication

Vol. 6, No. 1, January-March 2018, Serial Number 21

Research Institute for Information and Communication Technology
Iranian Association of Information and Communication Technology

Affiliated to: Academic Center for Education, Culture and Research (ACECR)

Print ISSN: 2322-1437
Online ISSN: 2345-2773
Publication License: 91/13216

Editorial Office Address: No.5, Saeedi Alley, Kalej Intersection., Enghelab Ave., Tehran, Iran,
P.O.Box: 13145-799 Tel: (+9821) 88930150 Fax: (+9821) 88930157
E-mail: info@jist.ir
URL: www.jist.ir

Indexed by:

- SCOPUS	www. Scopus.com
- Index Copernicus International	www.indexcopernicus.com
- Islamic World Science Citation Center (ISC)	www.isc.gov.ir
- Directory of open Access Journals	www.Doaj.org
- Scientific Information Database (SID)	www.sid.ir
- Regional Information Center for Science and Technology (RICeST)	www.ricest.ac.ir
- Iranian Magazines Databases	www.magiran.com

Publisher:
Regional Information Center for Science and Technology **(RICeST)**
Islamic World Science Citation Center **(ISC)**

This Journal is published under scientific support of
Advanced Information Systems (AIS) Research Group and
Digital & Signal Processing Research Group, ICTRC

Acknowledgement

JIST Editorial-Board would like to gratefully appreciate the following distinguished referees for spending their valuable time and expertise in reviewing the manuscripts and their constructive suggestions, which had a great impact on the enhancement of this issue of the JIST Journal.

(A-Z)

Ahmadi, Sadra, Shahid Beheshti University, Tehran, Iran
Ashrafi Payaman, Nosratali, Iran University of Science and Technology, Tehran, Iran
Derhami, Vali, Yazd University, Yazd, Iran
Ebrahimpour, Nader, Islamic Azad University Mahabad Branch, Mahabad, Iran
Ensan, Faezeh, Ferdowsi University of Mashhad, Mashhad, Iran
Farsi, Hasan, University of Birjand, Birjand, Iran
Fouladi, Kazim, University of Tehran, Tehran, Iran
Ghazvini, Mahdieh, Shahid Bahonar University of Kerman, Kerman, Iran
Izadkhah, Habib, Tabriz University, Tabriz, Iran
Kasaei, Shohreh, Sharif University of Technology, Tehran, Iran
Lorber, Florian, Alborg University, Alborg, Denmark
Mahdavi, Mehregan, Victoria University, Sydney, Australia
Majma, Mohammad Reza, Islamic Azad University Pardis Branch, Pardis, Iran
Mansoorizadeh, Muharram, Bu-Ali Sina University, Hamedan, Iran
Mavaddati, Samira, University of Mazandaran, Babolsar, Iran
Mirroshandel, SeyedAbolghasem, University of Guilan, Rasht, Iran
Mohammadi, Reza, Hamedan University of Technology, Hamedan, Iran
Mohanna, Shahram, University of Sistan and Baluchestan, Zahedan, Iran
Momtazi, Saeideh, Amirkabir University of Technology, Tehran, Iran
Mousavirad, Seyed Jalaleddin, University of Kashan, Kashan, Iran
Nilforoushan, Zahra, Kharazmi University, Tehran, Iran
Parsa, Saeed, Iran University of Technology, Tehran, Iran
Pour Ahmadi, Vahid, Amirkabir University of Technology, Tehran, Iran
Rahati, Amin, University of Sistan and Baluchestan, Zahedan, Iran
Ramezani, Reza, University of Isfahan, Isfahan, Iran
Rezvanian, Alireza, Amirkabir University of Technology, Tehran, Iran
Sadegh Mohammadi, Hamidreza, Iran University of Science and Technology, Tehran, Iran
Seyedin, Sanaz, Amirkabir University of Technology, Tehran, Iran
Shirvani Moghaddam, Shahriar, Shahid Rajaee Teacher Training University, Tehran, Iran
Yousefi, Saleh, Urmia University, Urmia, Iran

Table of Contents

Confidence Measure Estimation for Open Information Extraction

Vahideh Reshadat
Malek-Ashtar University of Technology, Tehran, Iran
vreshadat@mut.ac.ir
Maryam Hourali*
Malek-Ashtar University of Technology, Tehran, Iran
mhourali@mut.ac.ir
Heshaam Faili
School of Electrical and Computer Engineering, College of Engineering, University of Tehran, Tehran, Iran
hfaili@ut.ac.ir

Received: 29/Jul/2017 Revised: 17/Nov/2017 Accepted: 20/May/2018

Abstract

The prior relation extraction approaches were relation-specific and supervised, yielding new instances of relations known a priori. While effective, this model is not applicable in case when the number of relations is high or where the relations are not known a priori. Open Information Extraction (OIE) is a relation-independent extraction paradigm designed to extract relations directly from massive and heterogeneous corpora such as Web. One of the main challenges for an Open IE system is estimating the probability that its extracted relation is correct. A confidence measure shows that how an extracted relation is a correct instance of a relation among entities. This paper proposes a new method of confidence estimation for OIE called Relation Confidence Estimator for Open Information Extraction (RCE-OIE). It investigates the incorporation of some proposed features in assigning confidence metric using logistic regression. These features consider diverse lexical, syntactic and semantic knowledge and also some extraction properties such as number of distinct documents from which extractions are drawn, number of relation arguments and their types. We implemented proposed confidence measure on the Open IE systems' extractions and examined how it affects the performance of results. Evaluations show that incorporation of designed features is promising and the accuracy of our method is higher than the base methods while keeping almost the same performance as them. We also demonstrate how semantic information such as coherence measures can be used in feature-based confidence estimation of Open Relation Extraction (ORE) to further improve the performance.

Keywords: Information Extraction; Open Information Extraction; Relation Extraction; Knowledge Discovery; Fact Extraction.

1. Introduction

Information Extraction is the task of automatically extracting structured data from unstructured text. One of the core information extraction tasks is relation extraction which aims at extracting semantic relations among entities from natural language text. Relation extraction can potentially benefit a wide range of NLP tasks such as: Web search, question answering, ontology learning, summarization, building knowledge bases, etc. [1,2].

The huge and fast-growing scale, a mixed genre of documents and infinite types of relations are challenges of the Web-scale relation extraction [3]. The traditional approaches to information extraction (such as [4-6]) assume a fixed set of predefined target relations and usually don't scale to corpora where the number of target relations is very large [7,8]. An alternative paradigm, Open Information Extraction (OIE) aims to scale information extraction methods to the size and diversity of the Web corpus. Open IE systems extract relational tuples from text, without requiring a pre-specified vocabulary [9-12].

The key goals of Open IE are: (1) domain independence, (2) unsupervised extraction, and (3) scalability to large amounts of text [13]. Since Open IE is never perfectly accurate, it is helpful to have an effective measure of confidence.

Following [14], there are at least three important applications of accurate confidence estimation. First, accuracy-coverage trade-offs are a common way to improve data integrity in databases. Efficiently making these trade-offs requires an accurate prediction of correctness. Second, confidence estimates are essential for interactive information extraction, in which users may correct incorrectly extracted fields. These corrections are then automatically propagated in order to correct other mistakes in the same record. Directing the user to the least confident field allows the system to improve its performance with a minimum amount of user effort. Third, confidence estimates can improve performance of data mining algorithms that depend upon databases created by information extraction systems [15]. Confidence estimates, provide data mining applications with a richer set of "bottom-up" hypotheses, resulting in more accurate inferences.

* Corresponding Author

This paper focuses on the confidence estimation for Open IE systems. In this work we use logistic regression, a probabilistic machine learning model, to automatically assign a confidence weight to an extraction. This paper makes the following contributions:

- This paper proposes several diverse new lexical, syntactic and semantic features for estimating confidence of open relation extraction systems using a probabilistic model.
- We study how the proposed features for weighting extracted relations affect the performance of results and use a logistic regression classifier to assign a confidence score to each Open IE extraction in order to improve precision.
- Our evaluations show that the proposed method can drop noisy extractions from Open IE systems' outputs and demonstrate that effective incorporation of diverse features enables our approach to identify correct instances with more certainty.

The rest of this paper is organized as follows. Section 2 presents related work. Proposed methodology is described in Section 3. We present results of our experiments in Section 4 and end with conclusion and future work in Section 5.

2. Related Works

In this section we review some open information extraction systems with respect to confidence estimation.

WOE$_{pos}$ [16] applies Wikipedia for self-supervised learning of unlexicalized extractor and is limited to light features such as Part-Of-Speech (POS) tags. WOE$_{pos}$ generates relation-specific training examples by matching Infobox attribute values to corresponding sentences and abstracts these examples to relation-independent training data to learn an unlexicalized extractor. WOE$_{parse}$ [16] is a pattern classifier learned from dependency path patterns which uses typed dependencies as features. Authors in their evaluation showed that using deep syntactic parsing improves the precision of their system, however at a high cost in extraction speed.

R2A2 [17] exploits an argument learning component. It makes use of a number of classifiers to identify the arguments of a verb phrase (based on hand-labeled training data). Two classifiers identify the left and right bounds for first argument and one classifier identifies the right bound of second argument.

ZORE [18] is a syntax-based Chinese open relation extraction system for extracting semantic patterns and relations from Chinese text. ZORE realizes relation samples from automatically parsed dependency trees, and then extracts relations with their semantic patterns iteratively through a double propagation algorithm. [19] also considers Chinese Open relation extraction. It can be assumed as a pipeline of word segmentation, POS and parsing.

LSOE [20] is an Open IE extractor based on lexical-syntactic patterns. It provides a plain solution to perform rule-based extraction of facts using POS-tagged text. The method was developed based on two types of patterns: (1) generic patterns (2) rules from Cimiano and Wenderoth proposal [21]. LSOE performance was compared with ReVerb and DepOE. The results show that LSOE extracts relations that are not learned by other extractors and achieves compatible precision.

Wanderlust [22] uses hand-labeled training data to learn extraction patterns on the dependency tree. After annotating 10,000 sentences parsed with LinkGrammar, it learns 46 general linkpaths as patterns for relation extractions.

Some Open IE methods are designed to obtain binary facts and they usually don't capture higher order N-ary facts. KrakeN [23] considers this weakness. It can extract more facts per sentence in high precision and is capable of extracting unary, binary and higher order N-ary facts. Since using a dependency parser results in cost in recall and speed, many sentences were ignored due to heuristic of detecting erroneous parses. OLLIE [9] aims to improve the Open IE systems by using a hybrid approach based on bootstrapping. It learns pattern templates automatically from a training set that is bootstrapped from relations extracted by the ReVerb system. It obtains the pattern templates from the dependency path connecting pairs of entities and their corresponding relations. The patterns are then applied over the corpus and new facts are obtained.

ClauseIE [13] is a novel, clause-based approach to open information extraction which differs from previous approaches in that it separates the detection of "useful" pieces of information expressed in the sentence from their representation in terms of extractions. ClauseIE exploits linguistic knowledge about the grammar of the English language to first detect clauses in an input sentences and to subsequently identify the type of each clause according to the grammatical function of its constituents. ClauseIE attains high precision and recall and can be customized to output triples or n-ary facts. EXEMPLAR [24] is an ORE approach that extracts n-ary relations. It uses rules over dependency parse trees to detect relation instances. EXEMPLAR's rules are used to each candidate argument separately as opposed to all candidate arguments of an instance. Since the aim is to gain low computational cost and high precision, its variations have been indicated by different dependency parsers. The results are promising and EXEMPLAR outperforms the systems that support n-ary extraction.

Bast and Haussman [25] proposed a method called CSD-IE that uses contextual sentence decomposition for Open IE. It decomposes a sentence into the parts that semantically 'belong together'. The facts are then captured by recognizing the (implicit or explicit) verb in each part. In [26], the same authors improved the informativeness of extracted facts in Open IE by using some inference rules. Uninformative extracted facts are obstacle for semantic search applications utilizing them. Their evaluation shows that this approach can increase the number of correct and informative triples by 15% discarding the uninformative ones [27].

In [28] authors proposed an Open IE system based on semantic role labeling (SRL). They constructed novel extractors based on two semantic role labeling systems, one developed at UIUC's publicly available SRL system [29] and the other at LUND [30].

Existing Open Information Extraction systems have mainly focused on Web's heterogeneity rather than the Web's informality. The performance of the ReVerb system, drops dramatically as informality increases in Web documents. In [31] a Hybrid Ripple-Down Rules based Open Information Extraction (Hybrid RDROIE) system was proposed, which uses RDR on top of a conventional Open IE system. The Hybrid RDROIE system applies RDR's incremental learning technique as an add-on to the state-of-the-art ReVerb Open IE system to correct the performance degradation of ReVerb due to the Web's informality in a domain of interest. The Hybrid RDROIE system doubled ReVerb's performance in a domain of interest after two hours training.

We proposed two preliminary models called TR-DOE and RV-DOE [12]. These two kinds of hybrid systems are made of two shallow and deep Open IE systems by using two combination parameters separately. We detected the best trade-off between precision and recall. Experiments indicate that the proposed hybrid methods obtain significantly higher performance than their constituent systems. The best result was for TR-DOE which had an F-measure almost twice that of TextRunner.

Dependency-based Open information Extraction (DepOE) [32] is a multilingual OIE system based on fast dependency parsing which has the main feature of being able to operate at Web-scale. It uses DepPattern [33], a multilingual dependency-based parser, to analyse sentences and obtain fine-grained information. Then, a small set of extraction rules is applied and the target verb-based triples are generated. There is a more recent version of DepOE system, called ArgOE [11]. ArgOE is a multilingual rule-based OIE method that obtains as input dependency parses in the CoNLL-X format, recognizes argument structures within the dependency parses, and extracts a set of basic propositions from each argument structure. This method does not need training data and has higher recall and precision than previous approaches relying on training data.

Estimating a confidence score for Open Information systems is not addressed in literature so well. TextRunner [34], is first and high scalable Open IE system where the facts are assigned a probability. It counts the number of distinct sentences from which each extraction was found. Assessor uses these counts to assign a probability to each tuple using the probabilistic model.

ReVerb [27] is a strong and successful shallow Open IE system. It makes use of a simple POS tag sequence as a syntactic constraint in order to extract relation phrases and eliminate incoherent extractions and also reduce uninformative extractions. ReVerb uses a classifier to determine a confidence score for each triple. It employs a set of relation independent features and a training set containing 1,000 sentences from the Web and Wikipedia to assign a confidence score to each extraction.

OLLIE [9] is an Open IE system that learns pattern templates automatically from a training set that is bootstrapped from relations extracted by the ReVerb system. It uses a supervised classifier for confidence function. The classifier applies a set of lexical features such as frequency of the extraction patterns, position of function words etc.

Some related works to open relation extraction systems are semantic best-effort information extraction approaches. KnowItAll is a Web extraction system which labels its own training data. It aims to automate and simplify the process of extracting large scale relations from the Web. Its hypothesis is that extractions drawn more frequently from distinct sentences in a corpus are more likely to be correct.

In [14] authors showed conditional random field is an empirically sound confidence estimator for finite state information extraction systems. It has an average precision of 97.6% for estimation field correction. Scheffer et al. describes a confidence estimation algorithm using hidden Markov models in information systems in [35]. They estimate the confidence of only singleton tokens by the difference between the probabilities of their first and second most likely labels.

URNS [36] is a combinational “balls-and-runs” model that evaluates the impact of redundancy, sample size and corroboration from several distinct extraction rules on the confidence score. It was illustrated experimentally that the model's log likelihoods for unsupervised information extraction are considerably higher than previous methods.

In [37] Agichtein proposed an expectation-maximization algorithm for automatically assessing the quality of the extraction patterns and relation tuples for partially supervised relation extraction. This method was evaluated for different types of patterns and improved extraction accuracy over heuristic-based methods.

3. Proposed Method

In this section we describe our proposed approach for assigning probability of correctness to Open IE systems' extractions.

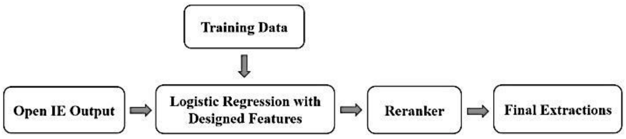


Fig. 1. The outline of Relation Confidence Estimator for Open Information Extraction (RCE-OIE)

There are various parameters that can aid in detecting accurate relations. This idea inspires us to develop a learning-based approach that applies our proposed parameters as features to assign a weight of correctness to the extracted semantic relations. Based on this assumption, the extractions with high precision will be obtained. The outline of the proposed method is shown in Figure 1.

Our proposed approach takes as input an output of Open IE system and trains a confidence metric on a labeled data set and uses the classifier's weight to assign a confidence score to each extraction.

3.1 Relation Confidence Estimator

Logistic regression is a conditional probability model which is used as the confidence classifier and is the main part of our approach. Relation Confidence Estimator for Open Information Extraction (RCE-OIE) reads every relation instance sequentially. For each relation instance, the confidence classifier computes the probability of its correctness.

This approach focuses on the confidence estimation for output instances of Open IE systems. We use logistic regression, a probabilistic machine learning model, to automatically assign a score to each input relation instance. Logistic regression belongs to the family of classifiers known as the exponential or log-linear classifiers. Like naive Bayes, it works by extracting some set of weighted features from the input, taking logs, and combining them linearly. It classifies an observation into one of two classes. In order to train a model to classify with minimum error as possible, the cost function should be minimized. Gradient descent is our learning algorithm that finds values for the parameters that result in best parameter values and a smaller minimum error. For this purpose we used Weka for implementation.

We formulate the relation confidence estimation for OIE systems as a classification problem by logistic regression classifier. Given the features and weights, our goal is to choose a class (confident or unconfident) for the relation instance. The probability of a particular class given the observation x is:

$$F(x) = \frac{1}{1+e^{-x}} \quad (1)$$

The logistic function maps values from -∞ and ∞ to lie between 0 and 1.

The problem addressed by the classifier is selecting proper class for each input relation instance in order to maximize the number of correctly extracted instances and thus effectiveness. This model can take our proposed features and return the probability that a particular observation is true and should be considered as a correct instance. We detect the best trade-off between effectiveness and efficiency (computational cost).

Deep features could improve precision and recall over shallow syntactic features, but at the cost of speed. For instance, parser-based features can help to recognize complicated and long distance relations in difficult sentences. Such cases usually cannot be detected by shallow features. Regarding the computational cost associated with rich syntactic features, we used about 14 light-weight features. All features are scalable, domain independent and can be evaluated at extraction time without use of expensive tools. These features could be extracted from the underlying systems.

3.2 Proposed Features

We designed various lexical, syntactic and semantic features to the classifier. All features are scalable, domain independent and can be evaluated at extraction time without use of expensive tools. These features are described in the following.

- Document frequency (D_f): This feature is based on the intuition that a valid relation phrase is found repeatedly in different documents in huge corpora such as the Web. More particularly, this feature considers redundancy impact on the probability of correctness and is defined as the number of distinct documents from which each extraction is found relative to the total number of documents.

$$Doc_f = \frac{|D_r|}{|D|} \quad (2)$$

$|D|$ is the total number of documents and $|D_r|$ is the number of documents containing relation r.

- Type frequency (T_f): This feature accounts for the number of domains in which the relation appears. We used Stanford NER for assigning types for arguments. It assigns one of the seven types (Location, Person, Organization, Money, Percent, Date, Time) to each argument. The arguments which are not in these classes are assigned "Other" tag. Let T={Location, Person, Organization, Money, Percent, Date, Time, Other}. Let domain type (DT) be the set of all possible relations' domain types (DT= T ×T). We use the frequency of domain types of arguments which reveals in the context of a relation. This feature is denoted as T_F and is defined as:

$$T_f = \frac{|AT_R|}{|DT|} \quad (3)$$

Where $|AT_R|$ is the number of distinct domain types that a relation takes and $|AT|$ is the total number of domain types.

Domain Entity frequency (DEf): This feature also considers the types of relation arguments and counts the number of distinct entity pairs of the type of relation's arguments which appear with it. This intuition is similar to that offered by Mesquita [38] to assign weights to the terms in the context of an entity pair in clustering task which could achieve high performance. In this case, we consider relation instead of term and define it as:

$$DE_f = \frac{|R_{at}|}{|R|} \quad (4)$$

$|R|$ is the frequency with which a relation r appears in the context of the arguments of any domain type. It shows the total number of occurrences of a relation with distinct arguments. $|R_{at}|$ is the frequency with which relation r appears in the context of the arguments of its current domain type. It is the total number of occurrences of a relation with distinct arguments of the type of its current arguments.

- Arguments frequency (A_f): This feature is based on the number of distinct arguments that a relation takes and is defined as:

$$A_f = \frac{|A_r|}{|A|} \quad (5)$$

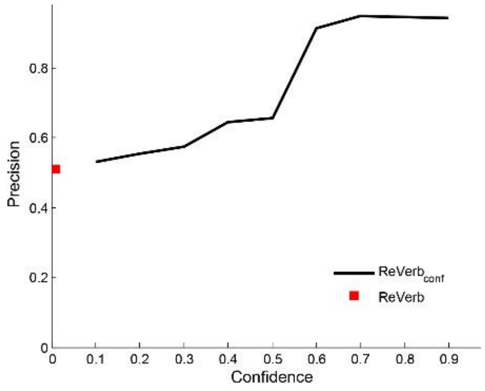
$|\mathrm{A}|$ is the total number of all distinct arguments in corpus and $|\mathrm{A_r}|$ is the count of distinct arguments which a relation takes.

- Arguments' Coherence: Coherence measures can be applied to automatically rate quality of topics computed by topic models [39]. A set of statements or facts is said to be coherent, if they support each other. We use the $\mathrm{C_v}$ and $\mathrm{C_a}$ measures proposed by Röder and his colleagues [39]. The framework of these coherence measures is a composition of four parts which differs in segmentation and probability calculation of words. One of the advantages of these measures is that they are based on word co-occurrence statistics estimated on Wikipedia and can detect coherence of proper nouns. Given two argument word sets, we calculate the coherence of each word in the first argument's word set with the words in the second argument's word set mutually and then compute the average of it. It measures the degree that two arguments are supported by each other.

We considered some syntactic and sentence-based features which are described in the follow.

- Arguments and relation covers all words in the sentence.
- There is a verb after second argument in the sentence.
- There is a preposition *in* or *to* after second argument in the sentence.
- First argument contains pronoun.
- There is a *in* or *if* before the first argument in the sentence.
- There is a *that* pronoun before the relation in the sentence.
- There is a *that* or *to* after the second argument in the sentence.
- Number of words in the sentence is less than ten.

We study how using these features for weighting extracted relations affects the precision of results. The next section gives more details about the results of our experiments.



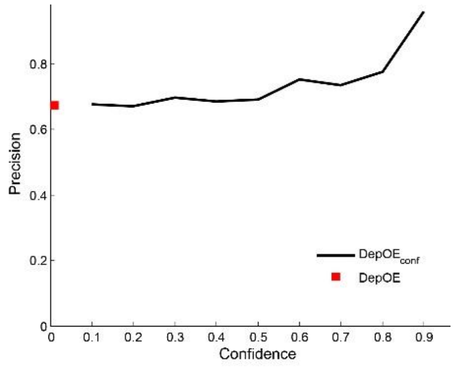


Fig. 2. ReVerb, DepOE and TextRunner have higher precision than their base systems for all thresholds.

4. Experiments and Results

In this section, we first describe benchmark datasets and performance metrics, and then give the results obtained by our approach and its counterparts.

4.1 Dataset

We used the dataset that was provided by Fader and his colleagues [27] in our experiments. They created a test set of 500 sentences sampled from the Web, using Yahoo's random link service. This dataset contains the output of the different extractors run (such as TextRunner and ReVerb) on the 500 selected sentences. Two human judges independently evaluated each extraction as 'correct' or 'incorrect'. The judges reached agreement on 86% of the extractions, with an agreement score of κ=0.68. The subset of the data where the two judges concur, is used in our experiments. The judges labeled uninformative extractions (where critical information was dropped from the extraction) as incorrect. This is a stricter standard than was used in previous Open IE evaluations [17].

In this collection, the extractions from a set of 1000 sentences from the Web and Wikipedia are available. The classifier was trained on 1000 random Web sentences with the proposed features. To collect syntactic features, we need to perform POS tagging and chunking therefore we use OpenNLP package [1]. Since the dataset only contains sentences, document frequency is estimated by assuming each sentence as a document.

4.2 Performance Measures

In the experiments, we conducted evaluations using two important criteria: precision and F-measure. For more detail about these metrics refer to [40]. The quality of the results is evaluated by comparing the relation instance pairs obtained by the system and those in the ground truth annotated by annotators. Formally, precision (P) and F-measure is defined as follows:

$$P = \frac{|S \cap G|}{|S|} \quad (6)$$

[1] http://opennlp.sourceforge.net

$$F - measure = 2 \times \frac{P \times R}{P + R} \qquad (7)$$

Where S is the set of relation instances generated by the system, and G is the set of correct labeled relation instances in the annotated gold standard set. R denotes recall, which is the ratio of the number of correct extractions retrieved to the total number of correct extractions in the dataset.

4.3 Experiment Results and Discussion

We evaluate the effect of applying logistic regression classifier, a linear regression method, on the output of different Open IE systems with the aid of some features and explore the behavior of it. We compare performance of TextRunner, ReVerb and DepOE with their confidence-based status.

A confidence score is assigned to each extraction using a classifier trained on mentioned training set with proposed features. Figure 2 reports the detailed precision curves of some Open IE systems with different confidence thresholds. Precision is the ratio of the number of correct extractions retrieved to the total number of extractions retrieved. The system names with *conf* subscript focuses on using only extractions with confidence values equal or above a threshold and ignores other extractions. As these figures show, precision curves always have higher levels of precision than their base for all confidence thresholds and all systems. This shows the effectiveness of proposed confidence score. Actually, the proposed method focuses on increasing the precision and uses the confidence as a filter policy to decrease the number of incorrect extractions and increase the correct ones, as a result, leads to the high precision.

DepOE's base system has higher level of precision than those of ReVerb and TextRunner. This is mainly because parser-based features used by DepOE are useful for handling correct extractions and thus, overall precision of it is high.

ReVerb and TextRunner start at low precision due to intrinsic weakness of shallow extractors in detecting relation instances.

Variations of precision for different values of confidence thresholds are also shown for all systems in Figure 2. When the confidence threshold is low, most of the extractions are considered as confident and the amount of precision for all systems is near the precision of their base cases. As confidence grows, the number of included extractions is gradually decreased but most of them are regarded correct therefore the precision slowly increases as confidence threshold increases. Figure 2 also shows that precision increases as confidence threshold increases but the slope of TextRunner and ReVerb's precision curves increase quicker than that of DepOE. Due to deep features used in DepOE, it extracts accurate triples and initial precision of it is higher than others and has relatively high start point difference with other approaches. It starts at high precision due to discarding of potentially low quality extractions from it. Thus, the proposed approach improves performance of both shallow and deep extractors. When confidence increases, the precision curves also increase as a result of filtering incorrect extractions.

The value of the threshold was examined from 0.1 to 0.9 by increments of 0.1. We examined F-measure values for all thresholds and found the maximum amount of it for each method. The results were shown in Figure 3. The F-measure is the uniformly weighted harmonic mean of the precision and the recall. Determination of the maximum value for F-measure is an attempt to find the best possible trade-off between recall and precision.

All systems achieve almost the same F-measure as their base. It shows that, the proposed method can achieve reasonable F-measure, but with more confidant extractions. Because of the deep tools used in the structure of deep_extractors, DepOE has the best F-measure in comparison with the other systems.

DepOE produces a little bit lower F-measures in comparison with its base case. The proposed method provides a boost in F-measures of the shallow extractors. TextRunner and ReVerb achieve an F-measure that is slightly higher than their base cases. This is mainly because of the depth of tools applied in their structures.

Features and learned weights in the logistic regression classifier are shown in Table 1. All of these features are effectively calculable and derived from corpus and sentences structures.

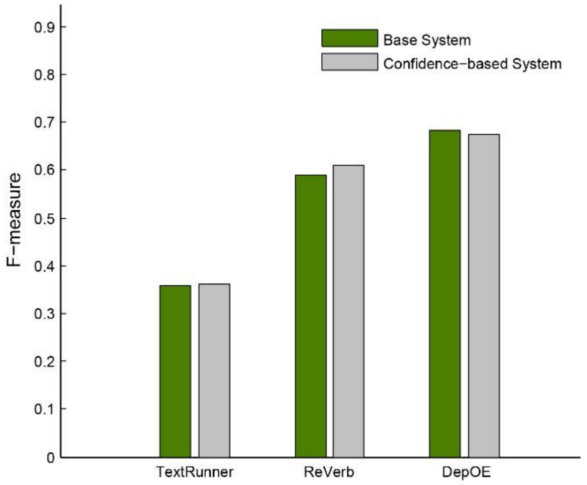


Fig. 3. All systems achieved approximately the same levels of F-measure

Since ReVerb as a robust and successful Open IE system is the nearest related work to our approach, we compare our method with it. ReVerb was applied on the test set of 500 sentences and the resulting extractions were used. We used Reverb's and our proposed confidence score and examined different threshold values to assess the precision variations. Our preliminary results from an analysis of ReVerb's output are reported in Figure 4.

The number of extractions with high confidence decreases as confidence threshold values increase and also the number of correct extractions increases as far as about all of retrieved extractions are correct in high confidence thresholds.

Table 1. Confidence classifier assigns a confidence score to an extraction from a sentence using these features.

Weight	Feature
0.01	D_f
0.12	T_f
0.24	DE_f
0.32	A_f
0.5	Cv
0.41	Ca
0.5	Arguments and relation covers all words in the sentence
0.49	There is a verb after second argument in the sentence.
-0.56	There is a preposition *in* or *to* after second argument in the sentence.
0.14	First argument contains pronoun.
-0.43	There is a *in* or *if* before the first argument in the sentence.
-0.61	There is a *that* pronoun before the relation in the sentence.
-0.42	There is a *that* or *to* after second argument in the sentence.
1.12	Number of words in the sentence is less than ten.

Except for extractions with a confidence values near 0.3, the precision of ReVerb$_{PROPconf}$ is always higher (or equal) than that of ReVerb$_{RVconf}$. This shows the effectiveness of proposed features. It seems that by increasing the number of effective features and the size of training data set, the results improves.

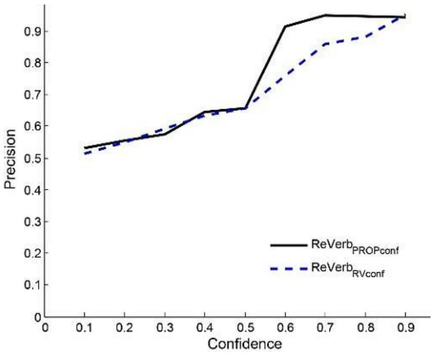


Fig. 4. Precision variation over different confidence values for ReVerb$_{PROPconf}$ and ReVerb$_{RVconf}$

4.4 Evaluating Classifiers

We modeled relation confident for OIE systems' outputs. Our modeling of relation confident was binary: relations are confident or unconfident. Given a corpus, proposed approach should select confident relations to maximize the number of correctly extracted instances.

Table 2 shows the distribution of the values in the confusion matrix for the confidence classifier. The results show that the dominant error in the classifier is classifying an unconfident extraction as confident.

Table 2. The confusion matrix for the performance of the confidence classifier

Gold/Classified	Confident	Unconfident
Confident	65.7%	23.9%
Unconfident	33.1%	77.3%

5. Conclusion

All of Open IE systems make errors and one of the important problems for an Open IE system is specifying the probability that extracted information is correct. In this paper, we used a logistic regression classifier to provide a confidence score for each relation of Open Information Extraction systems where diverse features are employed. It covers a wide range of features from syntactic and sematic (e.g., arguments' coherence) to sentence and corpus based ones (e.g. number of relation arguments and their type). Our evaluations show that effective incorporation of diverse features enables our approach outperform the base Open IE systems in terms of performance. Moreover, proposed features produce results comparable to the confidence score of ReVerb.

We plan to explore utilizing some more efficient features to improve performance of learned model. Furthermore, we are interested to extend experiments to other open IE systems and apply our model to their extractions. We also need to take into consideration the impact of training data set size and do experiments with larger amounts of training data to see if our new implementation improves. Another direction for improvement is to expand the type space of arguments with resources of semantic knowledge such as ontologies.

References

[1] L. Yao, A. Haghighi, S. Riedel, and A. McCallum, "Structured relation discovery using generative models," in Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2011, pp. 1456-1466.

[2] J. Piskorski and R. Yangarber, "Information extraction: Past, present and future," in Multi-source, Multilingual Information Extraction and Summarization, ed: Springer, 2013, pp. 23-49.

[3] B. Min, S. Shi, R. Grishman, and C.-Y. Lin, "Towards Large-Scale Unsupervised Relation Extraction from the Web," International Journal on Semantic Web and Information Systems (IJSWIS), vol. 8, pp. 1-23, 2012.

[4] R. C. Bunescu and R. J. Mooney, "A shortest path dependency kernel for relation extraction," in Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, 2005, pp. 724-731.

[5] A. Culotta, A. McCallum, and J. Betz, "Integrating probabilistic extraction models and data mining to discover relations and patterns in text," in Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, 2006, pp. 296-303.

[6] N. Kambhatla, "Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations," in Proceedings of the ACL 2004 on Interactive poster and demonstration sessions, 2004, p. 22.

[7] M. Banko, O. Etzioni, and T. Center, "The Tradeoffs Between Open and Traditional Relation Extraction," in ACL, 2008, pp. 28-36.

[8] C. C. Xavier, V. L. S. de Lima, and M. Souza, "Open information extraction based on lexical semantics," Journal of the Brazilian Computer Society, vol. 21, p. 4, 2015.

[9] M. Schmitz, R. Bart, S. Soderland, and O. Etzioni, "Open language learning for information extraction," in

Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, 2012, pp. 523-534.
[10] S. Soderland, B. Roof, B. Qin, S. Xu, and O. Etzioni, "Adapting open information extraction to domain-specific relations," AI Magazine, vol. 31, pp. 93-102, 2010.
[11] P. Gamallo and M. Garcia, "Multilingual open information extraction," in Portuguese Conference on Artificial Intelligence, 2015, pp. 711-722.
[12] V. Reshadat, M. Hoorali, and H. Faili, "A Hybrid Method for Open Information Extraction Based on Shallow and Deep Linguistic Analysis," Interdisciplinary Information Sciences, vol. 22, pp. 87-100, 2016.
[13] L. Del Corro and R. Gemulla, "ClausIE: clause-based open information extraction," in Proceedings of the 22nd international conference on World Wide Web, 2013, pp. 355-366.
[14] A. Culotta and A. McCallum, "Confidence estimation for information extraction," in Proceedings of HLT-NAACL 2004: Short Papers, 2004, pp. 109-112.
[15] A. McCallum and D. Jensen, "A note on the unification of information extraction and data mining using conditional-probability, relational models," 2003.
[16] F. Wu and D. S. Weld, "Open information extraction using Wikipedia," in Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, 2010, pp. 118-127.
[17] O. Etzioni, A. Fader, J. Christensen, S. Soderland, and M. Mausam, "Open Information Extraction: The Second Generation," in IJCAI, 2011, pp. 3-10.
[18] L. Qiu and Y. Zhang, "Zore: A syntax-based system for chinese open relation extraction," in Proceedings of EMNLP, 2014.
[19] Y.-H. Tseng, L.-H. Lee, S.-Y. Lin, B.-S. Liao, M.-J. Liu, H.-H. Chen, et al., "Chinese open relation extraction for knowledge acquisition," EACL 2014, p. 12, 2014.
[20] C. Castella Xavier, S. de Lima, V. Lúcia, and M. Souza, "Open information extraction based on lexical-syntactic patterns," in Intelligent Systems (BRACIS), 2013 Brazilian Conference on, 2013, pp. 189-194.
[21] P. Cimiano and J. Wenderoth, "Automatically learning qualia structures from the web," in Proceedings of the ACL-SIGLEX workshop on deep lexical acquisition, 2005, pp. 28-37.
[22] A. Akbik and J. Broß, "Wanderlust: Extracting semantic relations from natural language text using dependency grammar patterns," in WWW Workshop, 2009.
[23] A. Akbik and A. Löser, "Kraken: N-ary facts in open information extraction," in Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction, 2012, pp. 52-56.
[24] F. Mesquita, J. Schmidek, and D. Barbosa, "Effectiveness and efficiency of open relation extraction," Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, vol. 500, pp. 447–457, 2013.
[25] H. Bast and E. Haussmann, "Open information extraction via contextual sentence decomposition," in Semantic Computing (ICSC), 2013 IEEE Seventh International Conference on, 2013, pp. 154-159.
[26] H. Bast and E. Haussmann, "More informative open information extraction via simple inference," in Advances in information retrieval, ed: Springer, 2014, pp. 585-590.
[27] A. Fader, S. Soderland, and O. Etzioni, "Identifying relations for open information extraction," in Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2011, pp. 1535-1545.
[28] J. Christensen, S. Soderland, and O. Etzioni, "An analysis of open information extraction based on semantic role labeling," in Proceedings of the sixth international conference on Knowledge capture, 2011, pp. 113-120.
[29] V. Punyakanok, D. Roth, and W.-t. Yih, "The importance of syntactic parsing and inference in semantic role labeling," Computational Linguistics, vol. 34, pp. 257-287, 2008.
[30] R. Johansson and P. Nugues, "The effect of syntactic representation on semantic role labeling," in Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1, 2008, pp. 393-400.
[31] M. H. Kim and P. Compton, "Improving open information extraction for informal web documents with ripple-down rules," in Knowledge Management and Acquisition for Intelligent Systems, ed: Springer, 2012, pp. 160-174.
[32] P. Gamallo, M. Garcia, and S. Fernández-Lanza, "Dependency-based open information extraction," in Proceedings of the Joint Workshop on Unsupervised and Semi-Supervised Learning in NLP, 2012, pp. 10-18.
[33] P. G. Otero and I. G. López, "A grammatical formalism based on patterns of part of speech tags," International journal of corpus linguistics, vol. 16, pp. 45-71, 2011.
[34] M. Banko, M. J. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni, "Open information extraction for the web," in IJCAI, 2007, pp. 2670-2676.
[35] T. Scheffer, C. Decomain, and S. Wrobel, "Active hidden markov models for information extraction," in Advances in Intelligent Data Analysis, ed: Springer, 2001, pp. 309-318.
[36] D. Downey, O. Etzioni, and S. Soderland, "A probabilistic model of redundancy in information extraction," DTIC Document2006.
[37] E. Agichtein, "Confidence estimation methods for partially supervised relation extraction," in Proc. of SIAM Intl. Conf. on Data Mining (SDM06), 2006.
[38] F. Mesquita, "Clustering techniques for open relation extraction," in Proceedings of the on SIGMOD/PODS 2012 PhD Symposium, 2012, pp. 27-32.
[39] M. Röder, A. Both, and A. Hinneburg, "Exploring the space of topic coherence measures," in Proceedings of the eighth ACM international conference on Web search and data mining, 2015, pp. 399-408.
[40] D. M. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," 2011.

Vahideh Reshadat is a Ph.D. student of ICT Department of Malek-Ashtar University of Technology in Tehran, Iran. She obtained her B.Sc. degree in software engineering in 2008, and her M.Sc. degree in software engineering in 2011. Her main research area includes Natural Language Processing field and specially Information Extraction, Query Expansion and Ontology Learning.

Maryam Hourali was born in Shahrood, Iran. She received the Ph.D. degree from Tarbiat Modares University of Technology in 2012. Her main research area includes natural language processing field and specially text summarization, ontology learning and text analysis.

Heshaam Faili received his B.Sc. degree and M.Sc. degree in Software Engineering from Sharif University of Technology in 1997 and 1999 and his Ph.D. degree in Artificial Intelligence from the same university in 2006. At present, he is an Associate Professor of University of Tehran. His areas of research include Machine Intelligence and Robotics, Information Technology, Software.

Clustering for Reduction of Energy Consumption in Wireless Sensor Networks by AHP Method

Aziz Hanifi
Department of Management, Allameh Tabataba'i University, Tehran, Iran
hanifi_aziz@yahoo.com
Mohammad Reza Taghva*
Department of Management, Allameh Tabataba'i University, Tehran, Iran
taghva@atu.ac.ir
Robab Hamlbarani Haghi
Department of Mathematics, Payame Noor University, P.O. Box 19395-3697, Tehran, Iran
Robab.haghi@gmail.com
Kamran feizi
Department of Management, Allameh Tabataba'i University, Tehran, Iran
kamranfeizi@yahoo.com

Received: 11/Sep/2017 Revised: 22/Feb/2018 Accepted: 19/May/2018

Abstract

Due to the type of applications, wireless sensor nodes must always be energy efficient and small. Hence, some studies have been done in order to reduce energy consumption. One of the most important operations of wireless sensor networks is data collection. Due to the energy limitation of nodes, energy efficiency must be considered as an important objective in the design of sensor networks. In this study, we present a method in which nodes, in the first phase, find their position by using the position of the base station and two assumed nodes out of the environment where their geographical locations are known. In the second phase, we determine the cluster heads based on the criteria such as the remaining energy, the distance (the distance from the cluster head and the distance from the base station), the number of neighbors (the one-step neighbors and the two-step neighbors) and the centrality by using the multi- criteria decision making method. The proposed method in the NS2 environment is implemented and its effect is evaluated as well as compared with the NEECP E-LEACH protocols. Simulation results show that the proposed method improves the energy consumption, the network life span, the average packet delivery and the average delay.

Keywords: Clustering; Energy; Location; Base Station; Sensor Networks.

1. Introduction

The most important reason for the emergence and development of wireless sensor networks has been the continuous monitoring of contexts that are difficult or impossible to achieve by human beings [1]. In order to carry out the duties for a long time, these networks should be autonomous without the involvement of individuals. Such a network, according to its application, collects information about various events from its operating context and reports these information to the base station during initial processing, for instances, applications such as industry [4], crisis management [35,5], health [2,6] and military [3]. Each sensor node has a limited energy supply and in most applications, it is not possible to replace energy sources. Therefore, sensor node life is heavily dependent on the energy stored in its battery. Hence, extending the life span of such network is one of the most important issues [7,8].

Moreover one of the important issues of energy consumption in wireless sensor network is clustering. In the clustering method, the network is divided into a number of independent sets referred to as sets of clusters. So each cluster has a number of sensor nodes and cluster heads. The cluster nodes send their data to their cluster head node. The cluster head node aggregates the data and sends it to the base station. Therefore, clustering in sensor networks has some advantages such as supporting data aggregation, facilitating data collection, organizing an appropriate structure for scalable routing, and disseminating data efficiently over a network.

Research has proved that cluster head selected on single criterion doesn't have energy efficiency. Hence, an ideal cluster head is the one which is selected on multiple criteria. Solution of using multiple criteria can be solved via Multiple Criteria Decision Making (MCDM) technique. MCDM methods are used to solve the decision- making problem in engineering and sciences, with multiple attributes. MCDM techniques compare and rank multiple alternatives based on degree of desirability of their respective attributes. There are different types of MCDM approach. In this paper, it is used AHP (Analytic hierarchy process) method [32]. The AHP is a structured technique for organizing and analyzing complex decisions, based on mathematics and psychology. It was developed by Thomas L. Saaty in the 1970s and has been extensively studied and refined since then. It is used

* Corresponding Author

around the world in a wide variety of decision making situations, in fields such as government, business, industry, healthcare, shipbuilding and education [33].

In this paper, sensor nodes find their position by using the position of the base station and two other virtual points. Then by considering criteria such as remaining energy, the distance (distance from the cluster and distance from the base station), the number of neighbors (the one-step neighbors and the two-step neighbors) and the centrality, cluster heads are selected by using AHP method.

The rest of the paper is organized as follows: Section 2 reviews the related work. In Section 3, nodes, in the first phase, obtain their geographical position. In the second phase, cluster heads are selected by using AHP method. Section 4 analyzes experiments concerning existing nodes. Finally, in Section 5, we summarize discussion and conclude.

2. Review of the Related Literatures

Most of the clustering algorithms [9-12] have been established for WSNs according to heuristic methods. LEACH [9] as one of the popular distributed clustering algorithm where the sensor nodes designate LEACH offers substantial energy saving and extends the period of the network in comparison with the static clustering and minimum transmission energy. However, the chief difficulty of this algorithm is that there is a probability to choose a cluster head with very low energy, which may expire quickly and therefore reduces the performance of the network. Consequently, the amount of algorithms has been established to advance LEACH protocol, PEGASIS [13] and HEED [14] are prevalent among them. PEGASIS classifies the nodes into the chain in an attempt to make opportunity for each node to convey and obtain the data only from its neighbor nodes. In each turn, an arbitrarily designated node from the chain as a cluster head is chosen. PEGASIS is more efficient than LEACH; nevertheless, it is unbalanced for huge networks. Furthermore the delay is expressively high. Recently, many algorithms [15-19] have been established for data gathering structures for prolonging the lifetime of WSNs. Loscri et al. [20], have suggested TL-LEACH protocol presenting a novel level of hierarchy. It advances the network period over LEACH, however, with an extra overhead for selecting subordinate cluster heads and also non-cluster head nodes allocate to the cluster heads according to distance only, which may cause severe energy imbalance to the network. Xiaoyan et al. [22], have argued that M-LEACH algorithm is comparable with LEACH and only difference is that it forward to cluster head node in next hop rather than sending the data directly to the base station and thus it keeps energy in comparison with LEACH and TL-LEACH. However, in multi-hop data transfer between cluster heads, it does not regard the significant metrics like energy, node degree etc. Yassein et al. [21] discussed that V-LEACH protocol improves the LEACH protocol where some cluster heads referred as vice cluster heads are designated along with the chief cluster heads and once the main cluster heads die, the vice cluster heads play as a cluster heads. It is revealed that it acts better than unique LEACH. However, sensor nodes require additional processing energy for choosing cluster heads. Also, it doesn't mind of development of clusters, which may cause severe energy incompetence of the WSN. In [23], the researchers have argued that E-LEACH protocol, which is similar to LEACH protocol, but in the selection of cluster heads, remaining energy of the cluster heads was taken into consideration, which can spread the life of the network by saving the low energy of cluster heads. Means, it may not selects the cluster head with low energy. Bari et al. [24], have argued the least distance clustering (LDC) for enhancing the lifetime of WSNs. The value of their method is that it performs faster, due to the assigning of non- cluster head nodes to the nearest cluster head. The chief difficulty of LDC is the unsuitable creation of clusters. Nevertheless, it is problematic to discover the optimal clusters for large scale networks, since the computational difficulty differs exponentially. The studies [26-28], have planned energy well-organized cluster based routing arrangements for dependable networks and in [29] a framework for energy assessment in WSNs has proposed. A Novel Energy-Efficient Clustering Protocol (NEECP) is developed to improve the lifetime of sensor network. NEECP elects cluster heads in an influential manner and each cluster possesses various sensing range to balance the load on the cluster head. The protocol also uses the chain based data aggregation arrangements to spread the period of WSN. Furthermore, NEECP evades redundant data spreads that further advance the network lifetime. NEECP applies stochastic cluster head election procedure to select cluster heads [30].

3. Proposed Method

3.1 System Assumptions and Energy Model

Throughout the paper, assume that the following conditions hold:

- All nodes are constant and have resource constraints.
- The base station has no resource constraints.
- All nodes are located within one or more steps of the base station.
- All nodes have data to send at specific moments.
- The nodes are not equipped with a locator system.
- All of nodes that are located at the distance less than r meters from the base station, directly communicate with the base station (r is the radio radius of nodes).

Two nodes are located at the out of the environment where their geographic locations are known.

- There is not any guiding node in the network and all network nodes, using the proposed method of [43], can obtain their geographic location. The energy pattern as the same radio pattern in [31] is applied in this study. In this model, transmitter disperses energy to run the radio electronics and the power amplifier. The receiver dissipates energy to run the radio electronics. The energy consumption of the node depends on the amount of the data and distance to be sent. In this model, when the transmission distance d is less than the threshold distance d_0, the energy consumption of a node is relative to d^2, otherwise it is relative to d^4[31]. The whole energy consumption of each node in the network for conveying the k- bit data packet is given by the following equations.

$$E_{TX}(k,d) = E_{TX-elect}(k) + E_{TX-amp}(k,d) \tag{1}$$

$$= k\left(E_{elect} + E_{amp} d^{\alpha}\right) \tag{2}$$

$$= \begin{cases} k(E_{elec} + \varepsilon_{fs} d^2), & d < d_0 \\ k\left(E_{elec} + \varepsilon_{mp} d^4\right), & d \geq d_0 \end{cases} \tag{3}$$

Where the threshold distance d0 is:

$$d_0 = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}} \tag{4}$$

Where $E_{elect}$is the energy that is required for the run of the radio, ε_{mp} and ε_{fs} are the energy required to run the transmitter amplifier contingent on the distance d. To receive a k-bit message, energy consumed is:

$$E_{Rx}(k,d) = E_{Rx-ele}(k) = k.E_{elec} \tag{5}$$

Data collected from neighboring nodes are redundant and extremely connected. Therefore data is combined at the cluster heads. Energy dissipated for combining m messages of k bits is

$$E_{DA}(k) = E_{agg}.k.m \tag{6}$$

3.2 The Process of Determining the Geographic Location of Nodes

- According to Figure 1, we assume that one-step neighbors of the base station are aware of geographic locations of the base station and two assumed nodes outside the environment where their geographic locations are known. By this information, such a node can obtain its geographic position after calculating the distance between itself and the base station and two nodes outside the environment. For calculate distance between two nodes, there are different ways to estimate the distance between sensor nodes in WSNs such as RSSI, TOA, TDOA, RTOF and etc. Recently Adler and et al. presented a solution to get a precise estimation of the distance between two nodes without the needs for special purpose chips or a redesign of already existent nodes. It is used radio runtime measurement to calculate the distance between nodes and then it is presented algorithms to refine the measurements [37].

By continuing this method, all nodes are able to find locations.

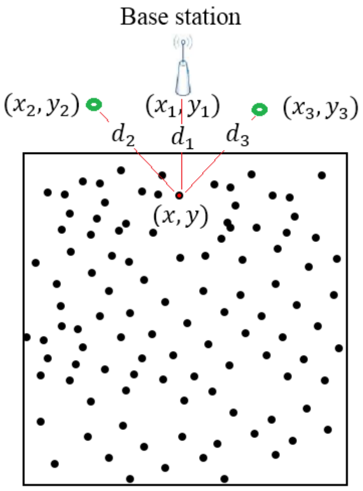


Fig. 1. Determining the geographical position of nodes

Now assume that (x, y) is the position of the unknown node which its distance from base station and two other known nodes with coordinates $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ are $d_{1,}d_{2,}d_3$ respectively. In this case the following equations hold

$$(x - x_1)^2 + (y - y_1)^2 = d_1^2 \tag{7}$$

$$(x - x_2)^2 + (y - y_2)^2 = d_2^2 \tag{8}$$

$$(x - x_3)^2 + (y - y_3)^2 = d_3^2 \tag{9}$$

So

$$x^2 + y^2 - 2xx_1 - 2yy_1 = d_1^2 + x_1^2 + y_1^2 \tag{10}$$

$$x^2 + y^2 - 2xx_2 - 2yy_2 = d_2^2 + x_2^2 + y_2^2 \tag{11}$$

$$x^2 + y^2 - 2xx_3 - 2yy_3 = d_3^2 + x_3^2 + y_3^2 \tag{12}$$

Equation (12) implies that

$$x^2 + y^2 = 2xx_3 - 2yy_3 + d_3^2 + x_2^2 + y_2^2 \tag{13}$$

Considering above equation and Equations 7 and 8,

$$\begin{cases} 2x(x_1 - x_3) + 2y(y_1 - y_3) = d_3^2 - d_1^2 + x_1^2 + y_1^2 - x_3^2 - y_3^2 \\ 2x(x_2 - x_3) + 2y(y_2 - y_3) = d_3^2 - d_2^2 + x_2^2 + y_2^2 - x_3^2 - y_3^2 \end{cases} \tag{14}$$

Finally it will be obtained two equations which have the solutions because it is assumed that the three points of $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ are not on the same line. Therefore the single-step neighbors of the base station obtain their geographical positions. Similarly two-step neighbors obtain their geographical positions using the geographical positions of single-step nodes. By continuing this way, after a while all nodes on the network receive their geographic location. The scheme of running the proposed method has been shown in Figure 2.

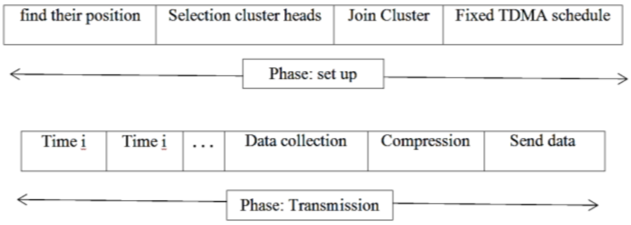


Fig. 2. The schemeof running the proposed method

3.3 Selecting Cluster Heads by AHP Method

Creating a hierarchical structure is one of criteria of service quality which aims to select optimally cluster heads based on the desired criteria. In the following, it is described the selection criteria for cluster heads with its reasons. In this section, the factors that are important in decision making are expressed as a hierarchical decision tree in the form of a decision tree, as shown in Figure 3.

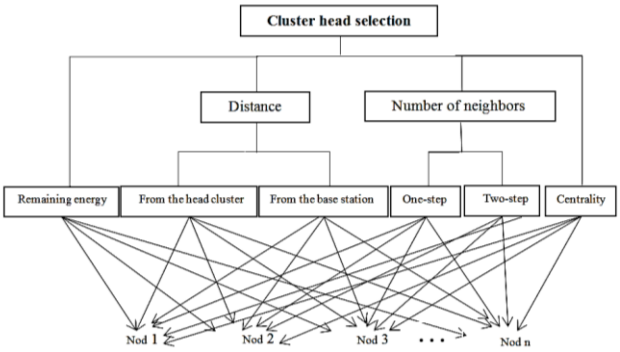


Fig. 3. Display of hierarchy to determine the cluster header

To determine cluster heads, Table 1 is utilized to converting linguistic expressions to numerical values.

Table 1. Numerical values of preferences in paired comparisons

Linguistic expression to determine preference	***numerical value***
Full Preference	9
Pretty strong	7
Strong preference	5
Little preference	3
The same preference	1
Preference between intervals	2,4,6,8

3.3.1 Criteria of Selection

- The distance from the base station: The more cluster head node is closer to the base station, the less the energy it uses to send the data packets. If the location coordinates of the node is (x_i, y_i) and the location coordinates of the base station is (x_j, y_j), then the distance of node from the base station is equal to:

$$d = \sqrt{((x_i - x_j)^2 + (y_i - y_j)^2} \tag{25}$$

- The distance from the cluster head: The closest node of cluster to the cluster head is the better candidate for cluster head. The distance of node from the cluster head is calculated similar to (25).
- The remaining energy of the node: Because the overhead of cluster head is larger than the other nodes, so the node should be selected as a cluster head that has enough energy, otherwise the nodes will be disconnected from the base station due to the node's death.

The number of neighbors: If r denotes the radio radius of each node, then the single- step neighbors of a node is the set of nodes that are located at a distance less than r meters from the node.

- The number of two-step neighbors: The two-step neighbors of the node are defined as the set of nodes which are located at the distance less than 2r meters from the node.
- Centrality: The mean distance of cluster nodes from a desired node is considered as centrality of that node. In fact reduction of centrality of the cluster head causes the energy consumption for intra-cluster communication (between nodes and cluster head).

If C is the set of cluster nodes, the centrality of the node x_0 is defined as follows:

$$\sum_{x_s \in C} \frac{|x_0 - x_s|}{|C|} \tag{26}$$

Where x_0 denotes the coordinate of the cluster node and x_s denotes the coordinate of the node within the cluster and $|C|$ is the number of nodes of C.

3.3.2 Determining the Paired Comparison Matrices

At this stage, the decision making matrix of the paired criteria is made. The ij^th entry of the decision-making matrix, in fact, is the ratio of the preference of i^th option to the j^th option. If the values are quantitative, it is sufficient to divide the values. If the values are qualitative, Table 1 is utilized to convert qualitative values to quantitative ones.

After making decision matrix, it should be checked the consistency or inconsistency of matrix. In the decision matrix, if the following equality is satisfied for all i, j, k,

$$a_{ij} = a_{ik}.a_{kj} \tag{27}$$

Then it is said that the matrix is consistent, otherwise is not.

3.3.3 Calculating Weights of Options and Criteria

In the following, the assumption of inconsistency of decision matrix is used to calculate the weights of criteria which have a very decisive role in decision-making problems which is in the form of Table 2.

Table 2. The paired comparison matrix of criteria

	remaining energy	**distance**	**number of neighbors**	**centrality**
remaining energy	1	a	b	c
distance	1/a	1	1/d	1/f
Number of neighbors	1/b	d	1	e
centrality	1/c	f	1/e	1

So the decision matrix is as follows:

$$A = \begin{pmatrix} 1 & a & b & c \\ \frac{1}{a} & 1 & \frac{1}{d} & \frac{1}{f} \\ \frac{1}{b} & d & 1 & e \\ \frac{1}{c} & f & \frac{1}{e} & 1 \end{pmatrix}$$

We use the eigenvector method to calculate weights. We obtain roots of the characteristic polynomial of matrix A, which is equal to solutions of equation $\det(kI -$

$A) = 0$. Assuming that k_{max} is the largest eigenvalue of A, the eigenvector associated with it is obtained by solving the Equation (28) using the MATLAB software. In this way, we obtain the corresponding eigenvector.

$$(k_{max}\,I - A)W = 0 \text{ where } \sum_{i=1}^{4} w_i = 1 \qquad (28)$$

Let corresponding eigenvector be (w_e, w_d, w_n, w_c). So the weights of criteria are obtained as follows:

The weight of the remaining energy = w_e

The weight of distance = w_d

The weight of the neighbor numbers = w_n

The weight of the centrality = w_c

For two criteria distance and the number of neighbors, below the paired comparison submatrices are made up.

Table 3. The paired comparison submatrices of distance

Distance	**Distance from the head cluster**	**Distance from the base station**
Distance from the head cluster	1	g
Distance from the base station	1/g	1

Table 4. The paired comparison submatrices of neighbors

The number of neighbors	**One-step neighbors**	**Two-step neighbors**
Two-step neighbors	**1**	**h**
Two-step neighbors	**1/h**	**1**

So the sub-matrices of the paired comparisons are as the following:

B_1 is the sub-matrix associated with the distance and

B_2 is the sub- matrix associated with the number of neighbors.

$$B_1 = \begin{bmatrix} 1 & g \\ \frac{1}{g} & 1 \end{bmatrix} \qquad B_2 = \begin{bmatrix} 1 & h \\ \frac{1}{h} & 1 \end{bmatrix}$$

Both matrices are compatible and the weights of each of the sub- criteria are as follows:

The weight of distance from cluster head = w_h

The weight of distance from base station = w_b

The weight of the number of single step neighbors = w_1

The weight of number of two-step neighbors = w_2

Now put:

$$E = \sum_{i=1}^{m} e_i \qquad (29)$$

$$N = \sum_{i=1}^{m} n_i \qquad (30)$$

$$M = \sum_{i=1}^{m} m_i \qquad (31)$$

$$D = \sum_{i=1}^{m} \frac{1}{d_i} \qquad (32)$$

$$D' = \sum_{i=1}^{m} \frac{1}{d'_i} \qquad (33)$$

$$C = \sum_{i=1}^{m} c_i \qquad (34)$$

Where

e_i: Remaining energy of ithnode

n_i: The number of one-step neighbors of i^{th} node in the cluster

m_i: The number of two-step neighbors of i^{th} node in the cluster

d_i: The distance of i^{th} node from the cluster head

d'_i: The distance of i^{th} node from the base station

c_i: The average distance of i^{th} node from all nodes of the cluster

For each node, calculate the following value, which is the final weight of the i^{th} node:

$$w_{\,node} = \frac{e_i}{E} w_e + \frac{1}{d_i D} w_d w_b + \frac{1}{d'_i D'} w_d w_h$$
$$+ \frac{n_i}{N} w_n w_1 + \frac{m_i}{M} w_n w_2 + \frac{c_i}{C} w_c \qquad (35)$$

Finally the node that has the biggest weight is selected as new cluster head.

Therefore, in the present paper, it has been tried to combine different criteria for choosing optimal cluster heads. Then we tried to provide the possibility of changing cluster head role after each transmission. So that the consumption of cluster energy is distributed equally among all cluster members and so early death of cluster nodes is prevented. The base station chooses cluster heads based on the criteria such as number of single-step and two steps neighbors, remaining energy, the distance from the cluster head, the distance from the base station and centrality of each node. For this purpose, a target function has been proposed that should be calculated for all nodes of each cluster. The base station calculates the weight w_{node} for each node, a node that has the biggest weight is selected as new cluster head. Finally the base station creates a packet which contains the geographic location of the nodes selected as cluster heads and then sends it to all nodes, so each node knows its cluster head.

4. Simulation Environment

The NS2 simulator is one of the most popular open source network simulators. For network research, NS is used as a discrete event simulator. The NS2 simulator is the second version of NS-Simulator. NS is essentially based on the network simulator called REAL. The original version of NS was designed in 1989. it has evolved in recent years and has continued to the third version. NS2's second version is widely used in academic research and has many packages that have been developed by people who have no financial benefit. Simulation on the Red hat Linux operating system using the NS2 network simulator was done.

Table 5. Simulation parameters used for WSNs

Parameter	***Value***
Number of sensor nodes	100
Target area	100*100
Energy of sensor node	2j
Transmission Range	30 m
Data Aggregation Energy	50pj/bit/report
Data Packet Size	4000 bits
Hello Packet Size	200 bits
Transmitter Electronics ($E_{elect}T_x$)	50 nJ/bit
Receiver Electronics ($E_{elec}R_x$)	50 nJ/bit
Transmit Amplifier (E_{amp})	100 pJ/bit/m2
Packet length	400 b
Transmission Frequency Band	2.4 GHz
MAC Protocol	CSMA/CA
Distribution	normal

In the first scenario, the base station is located in position at 100*100. In the second scenario, the base station is located in the optimal position.

Table 6. Simulation Scenarios

Base station	location
Scenario 1	100*100
Scenario 2	50*50

Evaluation Criteria:

To evaluate the efficiency of the proposed method, four criteria have been used: the average of consumable energy, the number of live nodes, the mean of end-to end delay, the mean of packet delivery success which refers to ratio of the number of packets delivered (successfully) to the base station to the number of packets generated by the source node. The four criteria are evaluated in two scenarios. In the first scenario, the location of the base station is placed at 100 * 100, and in the second scenario, the base station is located at the optimal position. In both scenarios, the four mentioned criteria have been evaluated. To evaluate the performance, the proposed method is compared with the methods of E-LEACH and NEECP methods.

Energy consumption of network per round is shown in Figure 4. It is observed that our proposed method consumes less energy than previous ones. The energy of the E-LEACH method ends at 440, and in the NEECP method, energy ends at 560, but in the proposed method energy ends at 760. A major constituent of energy consumption is communication process. So the proper communication model is very much necessary for any energy efficient clustering protocol. This is the main reason of lower energy consumption in our proposed method is much less than E-LEACH and NEECP method.

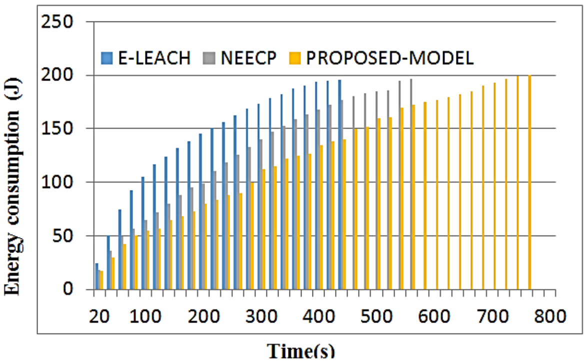


Fig. 4. Comparison of the amount of energy consumed in the first scenario.

Energy consumption of network per round in the second scenario is shown in Figure 5 It is observed that our proposed scheme consumes less energy than previous schemes. The energy of the E-LEACH protocol ends after 480 rounds, and in the NEECP method, energy ends after 600 rounds, but in the proposed method energy ends after 820 rounds. It follows that the proper location of the base station is effective on efficient energy consumption.

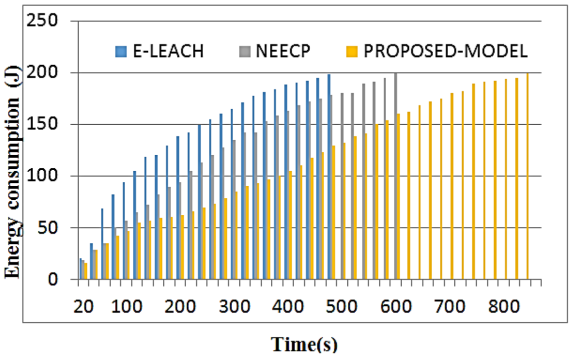


Fig. 5. Comparison of the amount of energy consumed in the second scenario.

In Figure 6, the lifetime of the network in the proposed method is compared with E-LEACH and NEECP protocols. In the E-LEACH algorithm, after 240 rounds, nodes start to dying vigorously, and after 380 rounds they are almost energized and the network lifetime ends. In the NEECP method, after 560 rounds, only 8 nodes are alive. While in the proposed method, network works with 32 live nodes after 560 rounds and the network lifetime ends after 680 rounds. The network lifetime of the proposed method is higher than the E-LEACH and NEECP methods. In the proposed method, the selection of the cluster heads due to the greater remained energy causes lower energy nodes not to participate in sending packets. So this is an important factor in increasing the lifetime of the network.

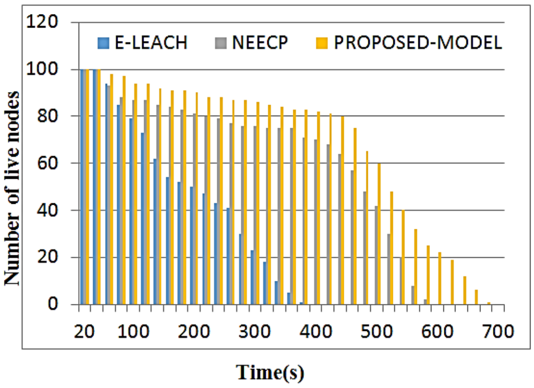


Fig. 6. Diagram of the number of live nodes (network lifetime) in the first scenario.

Figure 7 shows the network lifetime in the scenario that the base station is located optimally increases. In this figure, the lifetime of the network in the proposed method is compared with E-LEACH and NEECP methods. In the E-LEACH method, after 240 rounds, nodes start to dying vigorously, and after 380 rounds they are almost energized and the network life ends. In the NEECP method, after 560 rounds, only 24 nodes are alive. While in the proposed method, network works with 69 live nodes after 560 rounds and the network life ends after 740 rounds. The network lifetime of the proposed method is higher than the E-LEACH and NEECP methods because of optimally location of the base station. So the proper base station location is on efficient energy consumption.

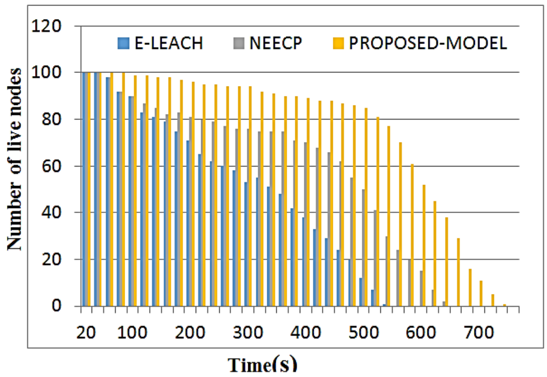


Fig. 7. the number of live nodes (network life span) in the second scenario.

Figure 8 shows that the proposed method minimizes end -to-end delay because of proper cluster heads selection and more density of nodes around the base station. Therefore, the delay is reduced compared to the NEECP and E-LEACH methods.

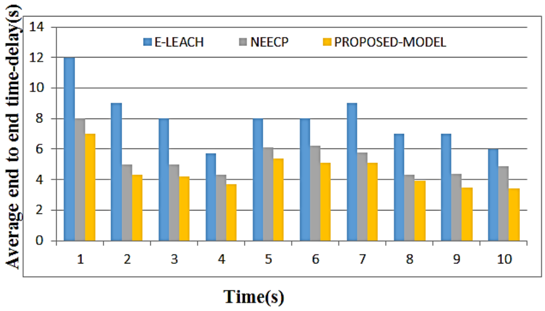


Fig. 8. The mean of end to end delay in the first scenario.

Figure 11 shows that the proposed method minimizes the end -to-end delay because the base station is located optimally and the nodes nearer to the base station need not firstly send the data to cluster head, they can directly communicate with the base station. Therefore, the delay is reduced compared to the NEECP and E-LEACH methods.

Figure 9 the first scenario, shows the percentage of accuracy of received packets which is yield by analyzing how many packets any node has sent, how many of them are actually received. As multi hop path is used for data communications, thus long distance communications is minimized so chance of data loss also is minimized.

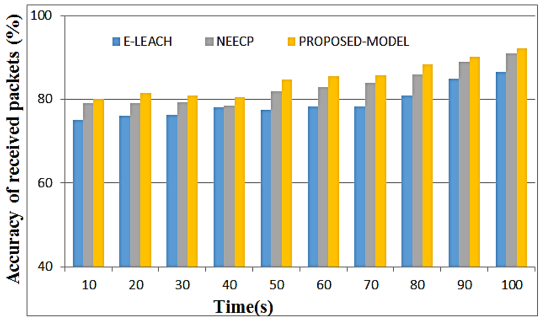


Fig. 9. Packets deliveries in the first scenario.

Figure 10 the second scenario in which the base station is located in optimal position shows the percentage of accuracy of received packets which is yield by analyzing how many packets any node has sent, how many of them are actually received. As multi hop path is used for data communication, thus long distance communications is minimized so chance of data loss also is minimized.

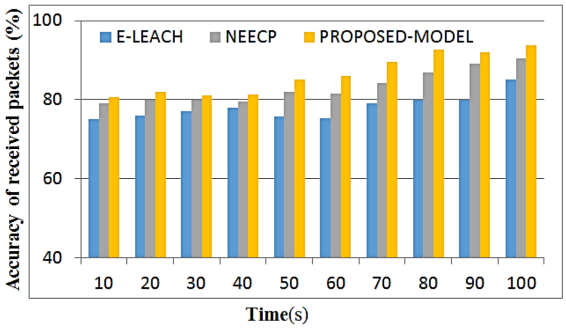


Fig. 10. Packets delivery in the second scenario.

5. Conclusion

In this study, we presented a method which has two phases. In the first phase, all sensor nodes obtain their geographical positions by using the location of the base station and two assumed nodes out of the environment where their geographic locations are known. In the second phase, the cluster heads are determined by the base station in such a way that the base station considered the remaining energy, the distance (the distance from the cluster and the distance from the base station), the number of neighbors (the one-step neighbors and the two-step neighbors) and the centrality as criteria for cluster heads selection by using AHP method. We simulated the proposed method by two scenarios. The first scenario assumed that the base station is located at position 100*100 and the second scenario assumed that it is located in 50*50. The proposed method has been compared to NEECP and E-LEACH methods in terms of energy consumption, life span, average delay and delivery. The simulation results showed that the proposed method had better performance because of choosing optimal cluster heads and base station location according to the density of nodes. One of the ways that can have a significant impact on reducing the energy consumption in WSN is determining the optimized location of the base station. Therefor the location of the base station has an important role in energy consumption of the network. So we will try to find a method for optimizing the location of the base station in the future.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasbramaniam, and E. Cayirci, "A Survey on Sensor Networks", In Proc. IEEE Communication magazine, 2002, Vol. 40, No. 8, pp .102-114 .

[2] J. Zheng, and A. Jamalipour, "Wireless sensor networks: a networking perspective", New Jersey: John Wiley & Sons, 2009.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey", Computer networks, Vol. 38, 2002, pp. 393-422.

[4] Q.Zhang, L. Yancheng, G. Haohao, and Q. Zhang "The Design of Hybrid MAC Protocol for Industry Monitoring System Based on WSN", Procedia Engineering, Vol. 23, No. 3, 2011, pp. 290-295.

[5] E. Cayirci, and T. Gupla, "Sendrom: Sensor networks for disaster relief operations management", Wireless Networks, Vol. 13, No. 3, 2007, pp. 409–423.

[6] Y.L. Zheng, "Unobtrusive Sensing and Wearable Devices for Health Informatics", IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, Vol. 61, No. 5, 2014, pp. 1538-1554.

[7] C -H. Tsai, and Y.-C. Tseng. "A path-connected-cluster wireless sensor network and its formation, addressing, and routing protocols." IEEE Sensors Journal, 2012, Vol. 12, No. 6, pp.2135-2144.

[8] I. F. Akyildiz, and M. C. Vuran, "Wireless sensor networks", United Kingdom: John Wiley & Sons, July 2010.

[9] A. Abbasi, and M. Younis, "A survey on clustering algorithms for wireless sensor networks", Computer communications, Vol. 30, No. 2, 2007, pp. 2826-2841.

[10] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "application-specific protocol architecture for wireless micro sensor networks," IEEE Trans. On Wireless Communications, Vol. 1, No. 4, 2002, pp.660–670.

[11] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," Proceedings of the 33rd Hawaii International Conference on System Sciences, 4-7 January 2000.

[12] L. Xiang, J. Luo, and A. Vasilakos, "Compressed data aggregation for energy efficient wireless sensor networks". In proc. 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2011, pp. 46-54.

[13] S. Lindsey, and C. S. Raghavendra, "PEGASIS: Power efficient gathering in sensor information systems". In proc. of IEEE Aerospace Conference, Vol. 3, 2002, pp. 1125–1130.

[14] O. Younis, and S. Fahmy, "HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks", IEEE Transactions on Mobile Computing, Vol.3 No.4, 2004, pp.366-379.

[15] Y.Yanjun, Q. Cao, A.V. Vasilakos, "EDAL: An energy-efficient, delayaware, and lifetime-balancing data collection protocol for heterogeneous wireless sensor networks", IEEE ACM Transactions on Networking, Vol. 23, No. 3, 2004, pp.810–823.

[16] Y. Liu, N. Xiong, Y. Zhao, A.V. Vasilakos, J. Gao, and Y. Jia "Multi-layer clustering routing algorithm for wireless vehicular sensor networks", IET Communications. Vol.4, No.7, 2010 , pp. 810–816.

[17] Y. Yao, Q. Cao, and A. V. Vasilakos, "EDAL: An Energy-Efficient, Delay-Aware, and Lifetime-Balancing Data Collection Protocol for Heterogeneous Wireless Sensor Networks", IEEE ACM Transactions on Networking, Vol. 23, No.3, pp. 810–823, 2015.

[18] K. Han, J. Luo, Y. Liu, and A. V. Vasilakos, "Algorithm design for data communications in duty-cycled wireless sensor networks: A survey", Communications Magazine, IEEE, Vol.51, No.7, 2013 , pp. 107–113.

[19] G. Wei, Y. Ling, B. Guo, B. Xiao, A. V. Vasilakos, "Prediction-based data aggregation in wireless sensor networks: Combining grey model and Kalman Filter", Computer Communications. Vol. 34, No.6, 2011 , pp. 793–802.

[20] V. Loscri, G. Morabito, and S. Marano, "A two-level hierarchy for low energy adaptive clustering hierarchy (TL-LEACH)," in proc. IEEE 62nd Vehicular Technology Conference, pp. 1809-1813.

[21] M. B. Yassein, A. Al-zou'bi, Y. Khamayseh, and W. Mardini, "Improvement on LEACH Protocol of Wireless Sensor Network (VLEACH)", International Journal of Digital Content Technology and its Applications, Vol. 3, No. 2, 2009, pp. 132-136.

[22] M. Xiaoyan, "Study and design on clustering routing protocols of wireless sensor networks", Ph.D Dissertation, Zhejiang University, Hangzhou, China, 2006.

[23] R. M. B. Hani, and A. A. Ijjeh, "A Survey on LEACH-based energy aware protocols for wireless sensor networks", Journal of Communications Vol. 8, No. 3, 2013.

[24] A. Bari, A. Jaekel, and S. Bandyopadhyay, "Clustering strategies for improving the lifetime of two-tiered sensor networks", Computer Communications, Vol. 31, No. 14, 2008, 3451–3459.

[25] N. Chilamkurti, S. Zeadally, A. Vasilakos, and V Sharma, "Cross-layer support for energy efficient routing in wireless sensor networks", Journal of Sensors, 2009, pp. 1-9.

[26] T. Meng, F. Wu, Z. yang, G. Chen, and A.V. Vasilakos, "Spatial reusability-aware routing in multihop wireless networks", IEEE Transactions on Computers, Vol. 65, No.1, pp. 244–255, 2016.

[27] P. Li, S. Guo, S. yu, and A.V. Vasilakos, "Reliable multicast with pipelined network coding using opportunistic feeding and routing" , IEEE Transactions on Parallel and Distributed Systems, vol. 25, no.12, 2014 , pp. 3264–3273.

[28] C. Busch, R. Kannan, A. V. Vasilakos "Approximating congestion + dilation in networks via ‘‘quality of routing games"", IEEE Transactions on Computers, Vol.61, No.9, 2012, pp. 1270–1283.

[29] N. Zhu, and A.V. Vasilakos, "A generic framework fo energy evaluation on wireless sensor networks", Wireless Networks. Vol. 22, No. 4, 2015, pp. 1199–1220.

[30] S. Singh, S. Chand, R. Kumar, A. Malik, B. Kumar, "NEECP: A Novel Energy Efficient Clustering Protocol for Prolonging Lifetime of WSNs", IET Wireless Sensor Systems, Vol. 6, No. 5, 151-157

[31] W. B. Heinzelman, A. P. Chandrakasan, H. Balakrishnan, "Anapplication-specific protocol architecture for wireless microsensor networks", IEEE Transactions on Wireless Communications, Vol. 1, No. 4, 2002, pp. 660–670.

[32] T. L. Saaty, "Decision making with the analytic hierarchy process", Int. J. Services Sciences, Vol. 1, No. 1, 2008.

[33] B.O. Saracoglu, "Selecting industrial investment locations in master plans of countries". European J. of Industrial

Engineering. Inderscience Enterprises Ltd, Vol. 7, No. 4, 2013, pp. 416–441.

[34] M. Taghva, A. Hanifi, K. feizi, and M. Taghavi- Fard, “Energy consumption management by clustering and localization of nodes in wireless sensor networks”. International Journal of Computer Science and Network Security, Vol. 17, No.16, 2017, pp. 273-377.

[35] M. Shakeri, S. M. Mazinani, “Crisis Management using Spatial Query Processing in Wireless Sensor Networks”. Journal of information system and telecommunication, Vol. 2, No . 6, 2017, pp. 97-110.

[36] J. V. Stoep, “Design and implementation of reliable Localization algorithms using received signal strength”, Master of Science in Electrical Engineering, University of Washington, 2009.

[37] S. Adler, S. Pfeiffer, H. Will, T. Hillebrandt and J. Schiller, “Measuring the distance between wireless sensor nodes with standard hardware”, Positioning Navigation and Communication (WPNC), 2012 9th Workshop on, Dresden, Germany, 15-16 March 2012

Aziz Hanifi is a Ph.D. student of IT management in Allameh Tabataba'i University in Tehran, Iran in. He received his B.Sc. in software Engineering in 2005, his M.Sc. in Information Technology in 2007. His area research interests include sensor networks, electronic commerce, IT governance.

Mohammad Reza Taghva is an Associated Professor of Industrial Management at Allameh Tabataba'i University in Tehran, Iran. He received his M.Sc. and Ph.D. from Université de Rennes I, France. His research interests include business intelligence, information systems, ERP post implementation issues, IT governance, and ITSM.

Robab Hamlbarani Haghi is an assistant professor of pure mathematics at Payame Noor university. She received the B.Sc., M.Sc. and Ph.D. degrees in pure mathematics from Azarbaijan Shahid Madani university in Tabriz, Iran. Her research interests include fixed point and approximation theories.

Kamran Feizi is a Professor Of Industrial Management in Allame Tabtaba'i Univ. Tehran , Iran.
He received his B.Sc. in Computer soft ware from Sharif Univ. Of Tech. 1974. Tehran Iran, M.Sc. in Operation Research from Southampton University UK 1977 and Ph.D. In Management from Southampton University UK 1990. His publications include 18 Books and 104 Scientific Articles

Concatenating Approach: Improving the Performance of Data Structure Implementation

Davud Mohammadpur
Faculty of Electrical and Computer Engineering, Malek Ashtar University of Technology
dmp@znu.ac.ir
Ali Mahjur*
Faculty of Electrical and Computer Engineering, Malek Ashtar University of Technology
mahjur@gmail.com

Received: 24/Sep/2017 Revised: 19/Jan/2018 Accepted: 08/Apr/2018

Abstract

Data structures are important parts of the programs. Most programs use a variety of data structures and quality of data structures excessively affects the quality of the applications. In current programming languages, they are defined by storing a reference to the data element in the data structure node. Some shortcomings of the current approach are limits in the performance of a data structure and poor mechanisms to handle key and hash attributes. These issues can be observed in the Java programming language which that dictates the programmer to use references to data element from the node. Clearly it is not an implementation mistake. It is a consequence of the Java paradigm which is common in almost all object-oriented programming languages. This paper introduces a new mechanism called access method, to implement a data structure efficiently which is based on the concatenating approach to data structure handling. In the concatenating approach, one memory block stores both the data element and the data structure node. According to the obtained results, the number of lines in the access method is reduced and reusability is increased. It builds data structure efficiently. Also it provides suitable mechanisms to handle key and hash attributes. Performance, simplicity, reusability and flexibility are the major features of the proposed approach.

Keywords: Programming Language; Data Structure Handling; High-Level Abstraction; Concatenating.

1. Introduction

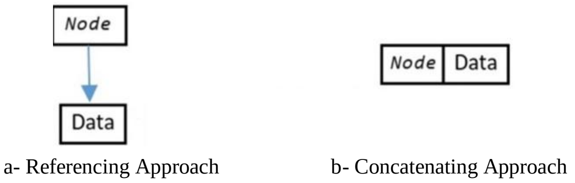
Data structures are important parts of programs. They are the building blocks of any program, and provide useful mechanisms to store and retrieve data [1]. Most programs use a variety of data structures. They often use simple variations or compositions of basic data structures such as linked lists, queues, stacks and tree types [2].

To illustrate some pervasive and serious problems in data structure management, we investigated data structures in many applications. For example, Hadoop, a distributed processing framework for large data sets, uses many Java data structures such as List, LinkedList, Queue, Set, TreeSet, LinkedHashSet, HashSet and HashMap. It can be concluded that, quality of data structures excessively affects the quality of the applications [3], [4].

Unfortunately, the usual approach to apply a data structure on a set of data elements is to store a reference to the data element in the data structure node (Fig. 1a) [5]. We call it referencing approach. In this approach, the data element and data structure node are allocated separately and the address of the data element is stored in the node. These references provide paths from structure nodes to data elements.

The referencing approach has two issues. First, it breaks an object into multiple parts (data element and data structure nodes). As stated in [6], breaking an object into multiple parts causes performance and memory penalties: ‘It incurs allocation and garbage collection overhead. Moreover, the fact that objects are accessed by reference introduces extra pointer dereferences. Finally, it incurs memory overhead: at a minimum, a pointer to the object and some memory for allocation administration is required’.

Second, there is no path from the data element to the corresponding data structure node. So, to reach the data structure node from the data element, the programmer has to scan the data structure. This increases the operations time, and limits performance on data structures.



a- Referencing Approach b- Concatenating Approach

Fig. 1. Data structure implementations

The better approach to apply a data structure on a set of data elements is to concatenate the data structure node to the data element (Fig. 1b). We call it concatenating approach. In this approach, one memory block stores both the data element and the data structure node.

This paper introduces a new mechanism called access method, to implement a data structure efficiently which is based on the concatenating approach. In this mechanism, a data structure is implemented independently. Later, programmers can apply data structures on data elements based on the concatenating approach.

* Corresponding Author

An important portion of data structures is the keys. In the access method mechanism, we define a special way to handle them. It allows a programmer to set a field(s) of the data element as a key.

2. Referencing Approach

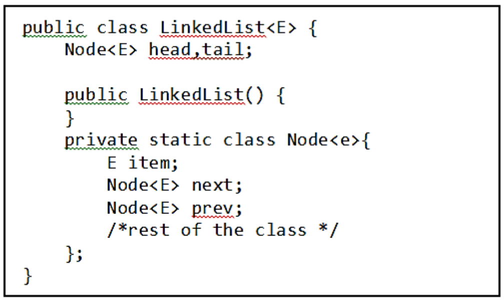
In current programming languages, to apply a data structure on a set of data elements, data elements are not stored in the data structure, but only references to the data elements are stored [7]. We call it referencing approach. As the Fig. 1a shows, the data element and data structure node are separated from each other, and the address of the data element is stored in the node. Therefore, two memory blocks are allocated per data element; one to store the data element and another one to store the data structure node [8].

It increases the memory footprint and reduces the performance of the code. The memory footprint is increased in two ways. One, as shown in Fig. 1a, some storage is used to store additional references in the data structure nodes. Two, dynamic memory management uses some extra storage to store its information. As this information is stored per block, increasing the number of blocks increases this overhead too.

The performance of the code is reduced due to the following reasons. One, two memory blocks should be allocated and freed, which increases the memory management time [9]. Second, it is not possible to reach a data structure node from its corresponding data element (Fig. 1a). The references only provide a path from the data structure node to the data elements. To find the corresponding data structure's node, the structure should be traversed which needs extra time [10].

As an example Fig. 2 shows an implementation of the doubly linked list in the current approach. It has two pointers: *head* and *tail*. *head* points to the first node of the list and *tail* points to the last node of the list. The node of the doubly linked list has two references: *next*, *prev* that point to other nodes. As is shown in Fig. 3 code snippet, removing a node from it needs O(1) time.

Even though removing a node from the linked list needs O(1) time, removing a data element from it needs O(n) time. To remove an arbitrary data element from the linked list, the programmer has to iterate over the linked list nodes to find the corresponding node and remove it [9]. Therefore, removing a data element from the linked list needs O(n) time.



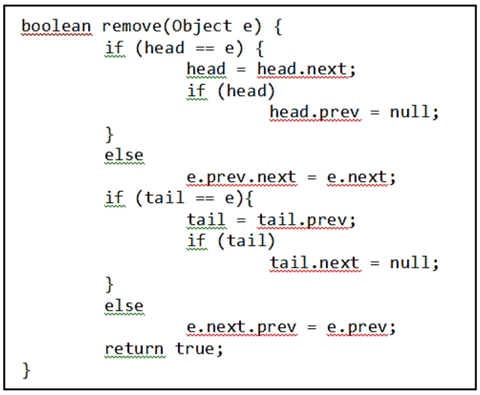
```
public class LinkedList<E> {
    Node<E> head,tail;

    public LinkedList() {
    }
    private static class Node<e>{
        E item;
        Node<E> next;
        Node<E> prev;
        /*rest of the class */
    };
}
```

Fig. 2. A Linked List

This issue can be observed in the Java *LinkedList*. The Java programming language dictates the programmer to use references to data element from the node. The following code snippet shows the node of the Java *LinkedList*. It has a field named *item* which points to the data element.

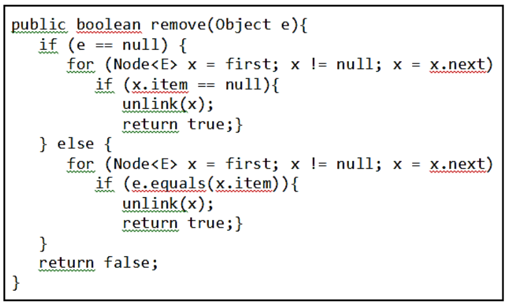
private static class Node<E>{
E item;
Node<E> next;
Node<E> prev;
};



```
boolean remove(Object e) {
        if (head == e) {
                head = head.next;
                if (head)
                        head.prev = null;
        }
        else
                e.prev.next = e.next;
        if (tail == e){
                tail = tail.prev;
                if (tail)
                        tail.next = null;
        }
        else
                e.next.prev = e.prev;
        return true;
}
```

Fig. 3. Removing A Node

To remove a node, it has *unlink()* method and to remove a data element it defines *remove()* method. *unlink()* needs O(1) time while *remove()* needs O(n) time. *remove()* is shown in the Fig. 4 code snippet. It traverse to locate the data element which needs O(n) time. After determining the corresponding node, the *unlink()* method is used to remove it.



```
public boolean remove(Object e){
   if (e == null) {
      for (Node<E> x = first; x != null; x = x.next)
         if (x.item == null){
            unlink(x);
            return true;}
   } else {
      for (Node<E> x = first; x != null; x = x.next)
         if (e.equals(x.item)){
            unlink(x);
            return true;}
   }
   return false;
}
```

Fig. 4. Removing A Data Element

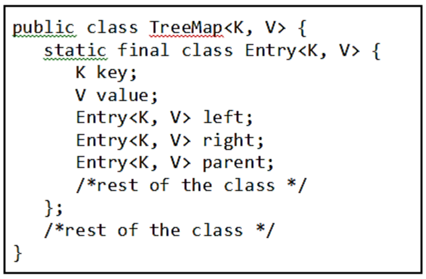
Clearly it is not an implementation mistake. It is a consequence of the Java paradigm which is common in almost all object-oriented programming languages.

Of-course C++ and non-object-oriented programming languages such as C lets programmer store the data element in the data structure's node, and thereby they are capable to alleviate the above issue. However in those languages to implement a data structure generally the programmer has to store a reference to the data element in the data structure's node. So, they have the same problem too.

Most data structures use a key to organize and retrieve data elements. The current approach to handle key is the key/value pair method [11]. As an example, Fig.5 is the

Java *TreeMap*. In this implementation, a parameter named *K*, is used as the data type of the key. In the *Entry* class, a new attribute, named *key*, is defined for internal storage of the key value. Also, a parameter named *V*, is used as the data type of the data element, and in the *Entry* class a new attribute, named *value*, is defined for internal storage of the data element. It should be considered that the key is a field(s) of the data element and can be extracted from it.

The mechanism has using additional storage for the key issue. Since the value of the key can be extracted from the data element, there is no need to store it. Moreover, the managing changes of the key value can lead to the redundancy.



```
public class TreeMap<K, V> {
   static final class Entry<K, V> {
      K key;
      V value;
      Entry<K, V> left;
      Entry<K, V> right;
      Entry<K, V> parent;
      /*rest of the class */
   };
   /*rest of the class */
}
```

Fig. 5. Java TreeMap

3. Access Method

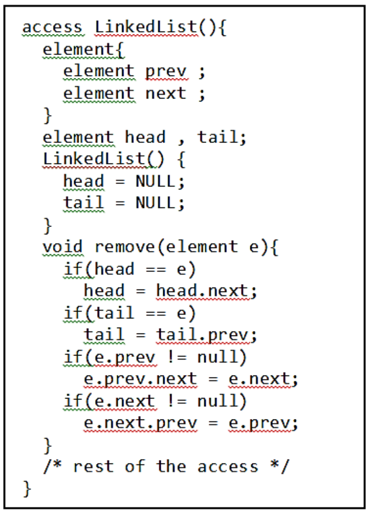
Examining data structures shows that their constructions follow the same framework. This framework has two segments. First, it has a segment, called node, which is responsible for keeping the main data. The node segment includes reference or references to other nodes along with the main data. The number and type of the references depend on the type of the data structure.

Second, for each data structure, a second segment, called root, is defined in which the general information of the structure is stored in. It is known as the input point to the structure. The root segment includes reference or references to some of the nodes of the structure. The management of the structure is implemented in the different operations in the root segment. The node segment usually does not perform separate operations except for providing the data [5]. We have presented the main idea called access method based on this common framework.

The access method is an abstraction for defining data structures. In this abstraction, the data structure is defined along with operations. For instance, the access method definition of a linked list is presented in Fig. 6.

In the implementation of the access method a section called element is used. The element points to the data structure of the node in the access method.

The element is the type too and points to the class of the node as a hypothetical data type. In this case, the element could be used as a data type for defining variables or in the definition of parameters. However, defined variables could not be allocated in any part of the access method. In fact, no part of the access method could get an independent memory. It is only allowed to point to the input memories.



```
access LinkedList(){
  element{
    element prev ;
    element next ;
  }
  element head , tail;
  LinkedList() {
    head = NULL;
    tail = NULL;
  }
  void remove(element e){
    if(head == e)
      head = head.next;
    if(tail == e)
      tail = tail.prev;
    if(e.prev != null)
      e.prev.next = e.next;
    if(e.next != null)
      e.next.prev = e.prev;
  }
  /* rest of the access */
}
```

Fig. 6. LinkedList Access Method

As shown in the code, the element section of the *LinkedList* has two attributes: *next* and prev. The defined access method for *LinkedList* includes one operator: *remove*. This operator acts on a variable of type *element*.

In the usage step, access method should be applied to a data element. By applying the access method on the data element, a new object is created, and the defined operations in the access method are provided along with the attributes and methods of the data element. The access methods could not be instantiated directly unlike conventional data structures. When an access method is applied, the created structure will include two segments: node and root. When an access method is applied to data, the element section is concatenated to the data, and the node segment is formed. The root segment consists of other attributes and operations, defined in the access method.

As an example, if class *Person* is defined as follows:

class Person{
int id;
string first_name;
string last_name;
string father_name;};

We could apply the *LinkedList* as shown below on the class *Person*. Thus, *people* will be a *LinkedList* of class *Person*.

Person[LinkedList()] people;

In the above example, the node object which is created for people by the compiler, includes two parts. The first part includes the defined items for class Person and the second part includes the defined items in the element section from the access method.

3.1 Key

An important characteristic of data structures is key values. To support key values, the access method has a special mechanism: hypothetical key type. If an access method has a key, it should define a key type. Inside an access method, the key type is like a usual data type. It can be used to declare variables and arguments. The only attribute of a key type is that it defines a linear order on the elements of the data set. Therefore, it is possible to compare two key values by their key.

Often it is required to extract the key of a data element. Assume that *e* is a data element that the key *k* is defined on it, *e.k* extracts it. As an example of key type, Fig. 7 code snippet has the definition of the binary search tree access method (*Tree*). It shows that the *Tree* access method has a key type named *k*. The *lookup* operation has an argument of type *k* and finds an element having that key, i.e. *e.k == ka*. Also, in the body of *insert* operation *k* is used to compare two key values, *e1.k < e2.k*.

An access method can have more than one key. As shown in the following code snippet, *k1* and *k2* are defined as two key types of *X*.

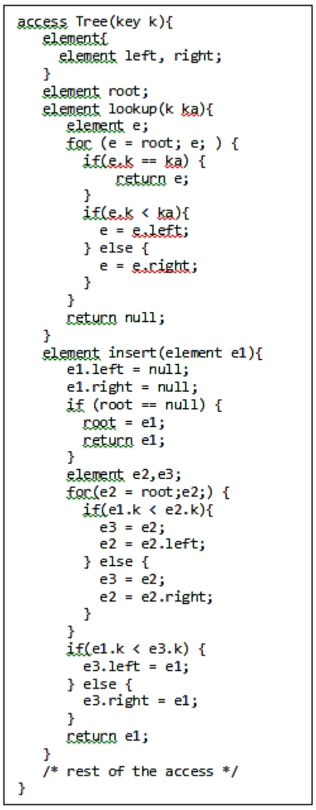
```
access X (key k1, key k2){
        /* rest of the access method */
}
```

When an access method is instantiated, its abstract key types should be assigned values. The value of a key is a sequence of expressions composed of the data element attributes and literals. The definition of an expression of a key type is embraced in a <> pair. Some examples of key definitions are followed (Person is the base type):

<id>
<lname, fname>

The first expression consists of one attribute and the second one consists of two attributes. For instance, applying the Tree access method can be done as follows:

Person[Tree(<id>)] people;



```
access Tree(key k){
   element{
     element left, right;
   }
   element root;
   element lookup(k ka){
      element e;
      for (e = root; e; ) {
        if(e.k == ka) {
            return e;
        }
        if(e.k < ka){
          e = e.left;
        } else {
          e = e.right;
        }
      }
      return null;
   }
   element insert(element e1){
      e1.left = null;
      e1.right = null;
      if (root == null) {
        root = e1;
        return e1;
      }
      element e2,e3;
      for(e2 = root;e2;) {
        if(e1.k < e2.k){
          e3 = e2;
          e2 = e2.left;
        } else {
          e3 = e2;
          e2 = e2.right;
        }
      }
      if(e1.k < e3.k) {
        e3.left = e1;
      } else {
        e3.right = e1;
      }
      return e1;
   }
   /* rest of the access */
}
```

Fig. 7. Tree Access Method

4. Translation into Java

The access method was implemented as an extension to the Java programming language. The compiler gets a code in the extended language and produces output in the Java language. The output can be compiled using any Java compiler to produce byte code. The compiler is implemented as a multi-pass translation in Java. The translation process is implemented by means of common tools such as JFlex and Cup. It includes three phases: lexical analysis, parsing and code generation (Fig. 8).

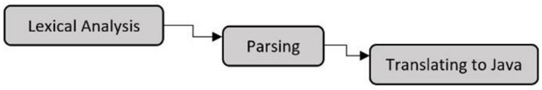


Fig. 8. Translation from the access method to Java

As specified in Fig. 8, in the first phase of lexical analysis and parsing, we perform syntactic checks like multiple declarations of the same named access methods, or declaration of element sections and operations. Next, if access method declaration and usage are matched, then the next step is the translation into Java. When we compile the back-end generated Java for execution, Type checking is handled in Java. The translation into Java is the most demanding step. During this phase, structural translation rules are followed to translate each class and access method into one or multiple classes. The resulting classes are then composed to build the complete Java representation of the source.

5. Results

As it was mentioned in the introduction, current programming languages use the referencing approach to apply a data structure on a set of data elements. The referencing approach has some issues. First, it increases the memory footprint, and second, it reduces the performance of the code. Now, the access method is implemented based on the concatenating approach, and it solves the issues of referencing approach.

To evaluate the access method, in this section it is compared with the Java and hand-coded implementations. In the Java implementation *LinkedList* and *TreeMap* is used from Java SE 10. As the time complexity of the Java approach is not satisfactory, the proposed data structures is implemented in hand-coded. In hand-coded implementation, data structures are implemented from scratch. This make more lines of codes than the access method implementation.

We perform testing for a variety of list sizes from 1000 items to 100M items. We use the Java Microbenchmark Harness (JMH) [12] test to conduct the test on a four core machine. The results are presented below subsections.

5.1 LinkedList

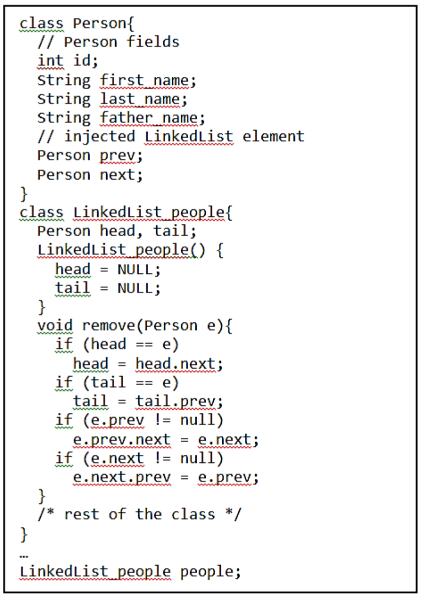
Assume that the *LinkedList* access method is implemented as is presented in Fig. 6. Consider the following code snippet, *LinkedList* access method is applied on class *Person*.

Person[LinkedList ()] people;

…

people.remove (p);

Fig. 9 shows the produced code for the above code snippet.



```
class Person{
  // Person fields
  int id;
  String first_name;
  String last_name;
  String father_name;
  // injected LinkedList element
  Person prev;
  Person next;
}
class LinkedList_people{
  Person head, tail;
  LinkedList_people() {
    head = NULL;
    tail = NULL;
  }
  void remove(Person e){
    if (head == e)
      head = head.next;
    if (tail == e)
      tail = tail.prev;
    if (e.prev != null)
      e.prev.next = e.next;
    if (e.next != null)
      e.next.prev = e.prev;
  }
  /* rest of the class */
}
…
LinkedList_people people;
```

Fig. 9. Produced Code for Applied Access Method

As noted before, element part of *LinkedList* is concatenated to class *Person* as data element. So, there is no need to additional references to operate on data structures. References to class *Person* are added to class *LinkedList_people* as root of data structure. Also *remove* method is customized and added to class *LinkedList_people* based on class *Person* as data element. As the data element and the node of data structure is concatenated together, so there is no need to scan data structure, and its *remove* operation be in O(1) time.

As mentioned, to evaluate the access method, we perform testing for the linked list. This test measures the performance of creating the linked list and populating the linked list for a specified number of items in the access method, Java, and hand-coded implementations. The test code is shown below. A specified number of integers is created using the Random class and collecting them into the linked list.

```
@State(Scope.Thread)
static public class MyState {
 @Param("1000")
 public int NSIZE;
}

@Benchmark
public void test_createLinkedList(MyState state) {
 Random random = new Random();
 LinkedList< Integer > list = random
  .ints(state.NSIZE)
  .collect(LinkedList::new, List::add, List::addAll);
}
```

The performance of the insert operation of the linked list is shown in Fig. 10 in the access method, the Java and hand-coded implementations. We tested from 1000 through 100M items as shown on the X-axis. The Y-axis is nanoseconds of an operation, and is shown in log scale since there is a slope up as the size increases in the java implementation.

In the next, the performance of the remove operation of the linked list is shown in Fig. 11 too.

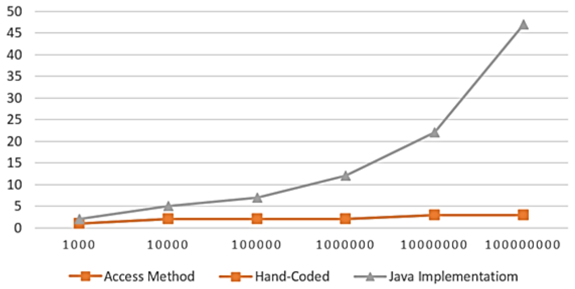


Fig. 10. Performance of the linked list insert operation in the access method, the Java and hand-coded implementations

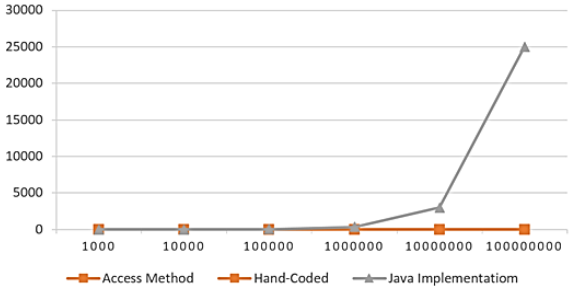


Fig. 11. Performance of the linked list remove operation in the access method, the Java and hand-coded implementations

5.2 Tree

The second test measures the performance of creating the tree and populating the tree for a specified number of items in the access method, the Java and hand-coded implementations. The test code is shown below. A specified number of integers is created using the Random class and collecting them into a particular type of tree in the access method, the Java and hand-coded implementations.

```
@State(Scope.Thread)
static public class MyState {
 @Param("1000")
 public int NSIZE;
}

@Benchmark
public void test_createTree(MyState state) {
 Random random = new Random();
 TreeMap < Integer, Integer > tree = random
  .ints(state.NSIZE)
  .collect(TreeMap::new, tree::add, tree::addAll);
}
```

The performance of the insert operation in the tree is shown in Fig. 12. As mentioned before, we tested from 1000 through 100M items as shown on the X-axis. The Y-axis is nanoseconds of an operation and is shown in log scale since there is a slope up as the size increases in the Java implementation.

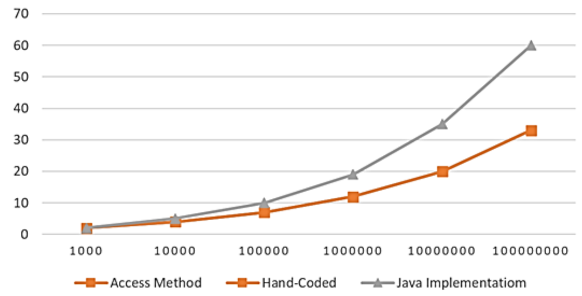


Fig. 12. Performance of the tree insert operation in the access method, the Java and hand-coded implementations

In the next, the performance of the remove operation in the tree is shown in Fig. 13.

According to the obtained results, the number of lines in the hand-coded implementation is high. Large volume of codes in the hand-coded approach makes it difficult to change and maintenance, and increase the complexity and cost of production. It's important to remember that hand-coded implementations are not reusable.

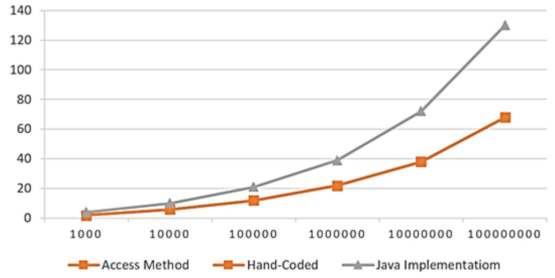


Fig. 13. Performance of the tree remove operation in the access method, the Java and hand-coded implementations

5.3 Discussion

The access method is similar to the hand-coded in time complexity. There's no perceptible difference between the access method and the hand-coded operations time. But, the number of lines in the access method implementations are low, and easy to reuse as the Java implementations. Since, the Java general data structures have high time complexity, and results show that as the number of items increases, they becomes slower which leads to lower efficiency compared to others.

6. Related Works

Several programming models attempted to provide high-level programming abstraction or interface in data structures. High-level programming models are in high-demand as they reduce the burdens of programmers [13]. However, the issue of the right high-level programming interface, especially in data structures, is not settled yet [14].

Rosenschein et al. [15] describe a language for specifying the requirements of a data structure. Then, the programming language selects the suitable data structure based on the specified requirements. Katz et al. [16] describe an expert system on data structures. The system is consulted by programmers during the design stage of their programs.

Schonberg et al. [17],[18] describe a technique for automatic selection of appropriate data representations during compile-time, and present a data structure selection algorithm in the SETL language.

Low [19] suggests that the data structures are represented as the abstract data types. For each abstract data type, some representations are provided, and the compiler chooses the best implementation.

7. Conclusions and Future Works

This paper introduced a new approach to implement data structures. The approach is based on four features: performance, simplicity, flexibility and not making any decision on behalf of the programmer. The approach consists of a new abstraction, the access method to define a data structure, and a new type for defining key. The provided samples show that the approach effectively reduces the cost of data structures operations and the approach creates a program-independent way to data structures define and manipulation.

The key direction for future work is extending the access method abstraction to support data structures compositions to provide the ability that an access method can make using other access methods.

References

[1] J. H. Drew, D. L. Evans, A. G. Glen, and L. M. Leemis, “Data Structures and Simple Algorithms,” in Computational Probability, Springer, 2017, pp. 89–109.

[2] I. Haller, A. Slowinska, and H. Bos, “Scalable data structure detection and classification for C/C++ binaries,” Empir. Softw. Eng., vol. 21, no. 3, pp. 778–810, 2016.

[3] M. Basios, L. Li, F. Wu, L. Kanthan, and E. T. Barr, “Optimising Darwinian Data Structures on Google Guava,” in International Symposium on Search Based Software Engineering, 2017, pp. 161–167.

[4] M. Basios, L. Li, F. Wu, L. Kanthan, D. Lawrence, and E. Barr, “Darwinian Data Structure Selection,” arXiv Prepr. arXiv1706.03232, 2017.

[5] T. H. CORMEN, C. E. LEISERSON, R. L. RIVEST, and C. STEIN, Introduction to Algorithms 3rd Edition. MIT press, 2009.

[6] C. Van Reeuwijk and H. J. Sips, “Adding tuples to Java: A study in lightweight data structures,” in Proceedings of the Concurrency Computation Practice and Experience, 2005, vol. 17, no. 5–6 SPEC. ISS., pp. 423–438.

[7] M. Sakkinen, "Disciplined Inheritance," in ECOOP 1989: European Conference on Object-Oriented Programming, 1989, pp. 39–57.

[8] Y. Zhang, M. C. Loring, G. Salvaneschi, B. Liskov, and A. C. Myers, "Lightweight, flexible object-oriented generics," in ACM SIGPLAN Notices, 2015, vol. 50, no. 6, pp. 436–445.

[9] S. Lindell, "A normal form for first-order logic over doubly-linked data structures," Int. J. Found. Comput. Sci., vol. 19, no. 1, pp. 205–217, 2008.

[10] C. Loncaric, E. Torlak, and M. D. Ernst, "Fast synthesis of fast collections," ACM SIGPLAN Not., vol. 51, no. 6, pp. 355–368, 2016.

[11] Y. Smaragdakis and D. S. Batory, "DiSTiL: A transformation library for data structures," in Proceedings of USENIX Conference on Domain-Specific Languages, 1997, no. October, p. 257270.

[12] Java.net, "JMH Test," 2017. [Online]. Available: http://openjdk.java.net/projects/code-tools/jmh/.

[13] N. Khammassi and J.-C. Le Lann, "A high-level programming model to ease pipeline parallelism expression on shared memory multicore architectures," Simul. Ser., vol. 46, no. 5, pp. 63–70, 2014.

[14] Y. Smaragdakis, "Technical Perspective High-Level Data Structures," Commun. ACM, vol. 55, no. 12, p. 2380656, 2012.

[15] S. J. Rosenschein and S. M. Katz, "Selection of representations for data structures," in Proceedings of the 1977 symposium on Artificial intelligence and programming languages., 1977, pp. 147–154.

[16] S. Katz and R. Zimmerman, "An advisory system for developing data representations," in Proceedings of the 7th international joint conference on Artificial intelligence, 1981, pp. 1030–1036.

[17] E. Schonberg, J. T. Schwartz, and M. Sharir., "An automatic technique for selection of data representations in setl programs," ACM Trans. Program. Lang. Syst., vol. 3, no. 2, pp. 126–143, 1981.

[18] E. Schonberg, J. T. Schwartz, and M. Sharir., "Automatic data structure selection in setl.," in Proceedings of the 6th ACM SIGACT-SIGPLAN symposium on Principles of programming languages, 1979, pp. 197–210.

[19] J. R. Low, "Automatic data structure selection: an example and overview," Commun. ACM, vol. 21, no. 5, pp. 376–385, 1978.

Davud Mohammadpur received his M.Sc. degree in Software Engineering from Iran University of Science and Technology. Currently he is a faculty member of University of Zanjan and a Ph.D. candidate at Malek-Ashtar University of Technology. His research interests are programming languages and information systems.

Ali Mahjur received his B.Sc., M.Sc. and Ph.D. from Sharif University of Technology. Currently he is a faculty member of Malek Ashtar University of Technology. His research interests are programming languages, operating systems and processor microarchitecture.

A New Set Covering Controller Placement Problem Model for Large Scale SDNs

Ahmad Jalili*
Department of Computer Engineering & IT, Shiraz University of Technology, Shiraz, Iran
a.jalili@sutech.ac.ir
Manijeh Keshtgari
Department of Computer Engineering & IT, Shiraz University of Technology, Shiraz, Iran
ahja2005@gmail.com
Reza Akbari
Department of Computer Engineering & IT, Shiraz University of Technology, Shiraz, Iran
Jalili_ah@yahoo.com

Received: 20/Oct/2017 Revised: 07/Jan/2018 Accepted: 19/May/2018

Abstract

Software Defined Network (SDN) is an emerging architecture that can overcome the challenges facing traditional networks. SDN enables administrator/operator to build a simpler and manageable network. New SDN paradigms are encouraged to deploy multiple (rather than centralized) controllers to monitor the entire network. The controller placement problem is one of the key issues in SDN that affect all its aspects including scalability, convergence time, fault tolerance, and node to controller latency. Many researchers focus on solving this problem by trying to optimize the location of an arbitrary number of controllers. The related works in this area get less attention to two following important issues: i) Bidirectional end-to-end latency between the switch and its controller instead of propagation latency and ii) finding the minimal number of controllers, which is a prerequisite for locating them. In this paper, we propose a Set Covering Controller Placement Problem Model (SCCPPM) in order to find the least number of required controllers with respect to carrier-grade latency requirement. The proposed model is carried out on a set of 124 graphs from the Internet Topology Zoo and solve them with IBM ILOG CPLEX Optimization package. Our results indicate that the number of required controllers for high resilient network is dependent on topology and network size. Moreover, to achieve carrier-grade requirement, 86% of topologies must have more than one controller.

Keywords: Software Defined Networks; Controller Placement Problem, Latency Constraint, Carrier-Grade Requirement.

1. Introduction

Software Defined Networks (SDN) is an emerging approach for dealing with the rigidity of traditional network. Unlike traditional networks that both data and control planes are tightly coupled on the same boxes, data and control planes are de-coupled [1]. In SDN, the complexity of data plan rules is off-loaded to the external intelligent modules called controllers. Such separation architecture enables administrator/operator to build a customizable, manageable, adaptable, and simpler network. Many efforts have been made on this separable architecture, which has diverse applications in the realm of data centers (DCNs), cellular networks, cloud, internet of things (IOT), wireless networks, and so on [1].

Recently, a substantial attention has been paid on the SDN concepts extending into wide area networks (WAN) and carrier networks [2]. Utilizing the advantages of logically centralized control of this architecture, it is possible for Carrier Network Infrastructure/WAN organizations to simplify and optimize the management of their network.

Although SDN has wide applications, it suffers from the inherent challenges, which should be probed such as scalability, reliability, and resiliency, specifically in WANs and carrier-grade networks [3].

Todays, WAN/carrier technologies are facing a rapid growth that provides remarkable characteristics and benefits like high availability, high resiliency, scalability, and reliability. For failure recovery, some networks offer carrier-grade quality meaning that a network should recover from failures within 50 ms. For instance, SONET/SDH has a specific protection strategy to provide high availability of service and they can achieve restoration time after failure at the order of 50 ms [4]. Achieving a high resilient communication is one of the major goals of networking. As a replacement for other well-established technologies, SDNs per se are expected to yield the same levels of resiliency as legacy alternative technologies in WAN.

Indeed, SDN must meet the resiliency and availability requirements of today's production networks to be a reliable alternative to the traditional network architecture.

Resiliency is defined as the persistence of service delivery that can defensibly be trusted when facing changes [5]. In a similar fashion, resilience in SDN is the capability to return to a previous state after the occurrence of some events or actions, which may have changed that

* Corresponding Author

state. In broad terms, an event in SDN implies any kind of occurrence that changes the state of the network to an unstable state such as unexpected failures, arriving a new flow, attacks and any variation in Qos parameters. When an event occurs or an SDN element is faulty, the network should provide a continuous operational service with the same performance [6]. Due to the separation of control and data planes, the issues related to resiliency are more challenging in SDN.

The main reason for this issue is the resilience of such network that depends on fault-tolerance in the data plane (as in traditional networks) and on the high availability of the control plane functions [7].

Furthermore, in SDN, switches are simple and passive devices that cannot perform any computations on their own. Therefore, the distance between the switch and its assigned controller impose more delay on restoration operations causing increasing recovery time.

To elucidate this subject, a scenario is presented to depict what happens in the network whenever an event occurs. When a switch detects an event[1], a notification message is sent to the controller, which then takes the required actions and installs updated flow entries in the required switches to conduct the incident occurred. Such reactive strategies imply high restoration time due to the necessary interaction with the controller. One experimental work on Open Flow for carrier-grade networks investigated the restoration process and measured a restoration time in the order of 100 ms [8].

The delay introduced by the controller, intermediate switches, and load of the network may be prohibitive, especially when in-band control is in play.

Generally, various metrics affect the resilience and restoration time. Some of these metrics include performance and resilience of control and data planes, the processing power of controller, the platform of the switch, the distance between switch and controller, recovery scheme, controller workload, and so on. In the literature, various ideas have been proposed to achieve high resiliency and improve restoration time. For instance, compressing the packet in messages [9], delegating some control decisions to the forwarding devices [10], protection schemes [8], designing and deploying high-performance controllers [11] [12], and the distributed control plane [13] [14] are elaborated in Section II.

Here, we start to look at the problem of resiliency in SDNs from a different perspective. As previously mentioned, the distance between switches and their assigned controller affects restoration time and resiliency, especially for wide area SDN deployments. Furthermore, one of the most important problems in SDN is controller placement problem (CPP), which is defined as how many controllers needed and where they should be placed to satisfy the optimal network performance [15] [16].

There are many papers published on CPP. Heller et al. established the first study in this area. In addition, many papers have been published trying to extend this work [15]. Using a brute-force method, authors evaluated the impact of controller placement on average and maximum latency metrics for real network topologies. Unsurprisingly, they show that in most topologies one single controller is enough to fulfill “existing reaction-time requirements”. In this work, in order to measure latency between switches and controllers, only propagation latency is considered. As well, the other following works also continue this imperfect scheme, except that they consider other objectives such as latency between controllers and load balancing.

Generally, the related works in this area have received less attention with respect to the following two issues. i) In order to calculate the distance between switches and controllers or between controllers, they only consider propagation latency in their measurement; while, this metric is widely variable and it depends on many parameters. We delve more deeply into this matter in Section III-A. Moreover, we show how to calculate end-to-end latency between switches and controllers. ii) They focus more on optimizing the location of an arbitrary number of controllers to satisfy some objectives such as load balancing, latency and so on.

They do not argue the number of controllers and the reason for its selection. Meanwhile, finding the required number of controllers with respect to resiliency is of highest importance especially in the case of distributed control planes; because if you do not know how many controllers you need, you cannot locate them optimally. Indeed, determining the number of required controllers is a prerequisite matter for locating them.

To find an optimum number of controllers, several strategies may be considered. 1) Since the main goal of architectures based on SDN is centralizing the control plane, at the first glance one controller is enough [15] [17].

However, fully physically centralized control is inadequate because it limits (i) responsiveness, (ii) reliability, and (iii) scalability. 2) In the second strategy, each switch is allocated to a dedicated controller that is directly attached to it. Although this way reduces latency and response time very much, it is not effective at all; as it is the same as the architecture of traditional networks with their inherent challenges [4]. 3) The last strategy suggests determining the number of required controllers in some network by a specific scheme (in section III-B). Nevertheless, it is to be noted that the number of required controllers depends on various constraints regarding the requirement of network administration. This is not to say that there are many requirements in the network, which can be considered as constraints such as latency, Qos metrics, administration aspects, load, and various domains in autonomous systems. In this strategy, some essential constraints are considered based on the defined requirements of the network, and then calculations are practiced to find the least number of controllers.

[1] In this paper, an event implies any variation or any observable occurrence in the network such as arriving a new flow, congestion detection, link failures, controller failures, TCP timeouts, port-down event, adjacency switch failure and any variation in traffic parameters.

Here, only latency constraint is considered in the modeling, due to focusing on restoration time as a most important requirement in network resiliency. In order to meet carrier grade requirements, this constraint must be up to 50 ms. Our analysis is based on the condition that whenever an event occurs within the network there is tens of mili-seconds opportunity for the network to return to the proper state (or to handle it). Furthermore, in order to yield the same levels of resiliency as carrier technologies, it is assumed this time must not exceed 50 ms.

For this purpose, a Set Covering Controller Placement Problem Model (SCCPPM) is proposed to determine the least number of required controllers regarding carrier grade latency requirement. The new model is carried out on a set of 124 graphs from the Internet Topology Zoo, which are solved using the IBM ILOG CPLEX Optimization package.

As expected, the minimum number of required controllers varies greatly and is more related to the network topology than the network size. Besides, in order to achieve carrier-grade requirement, 86% of topologies must have more than one controller. However, we absolutely need more than 86% of topologies in case of consideration of some other administration constraints.

Finally, we conclude this paper by discussing the main results of our analysis, which indicates that resilience in SDNs is achievable by carefully choosing the number of controllers within the target network topology.

The proposed model is of great importance from several aspects: i) By this mathematical model, decision makers and authors can determine the minimal number of controllers for their specific networks with any kind of structures or any size. They do not need to arbitrary choose or estimate it anymore. ii) Because of diverse requirements in networking, this model can be applied to the various networks regarding defined constraints for them. iii) With acquiring the number of controllers required, it is possible to estimate the cost allocation of designing and deploying of a specific SDN.

Therefore, it is necessary to know many controllers we need. Some papers have introduced the distributed control plane but they have not investigated the amount of the minimum number of controllers. In summary, in the present work, the following contributions are made:

- To the best of our knowledge, this is the first study that illustrates the concept of restoration time regarding bidirectional end-to-end delay between switches and controllers.
- The mathematical model proposed in this work first evaluates the resiliency constraint, such that it can be considered as a position paper for other works that have been addressing CPP. For this purpose, initially, the least number of controllers is determined through the proposed model. Then, in the next phase, utilizing the other approaches in the literature, controllers are optimally deployed in the network.

The remainder of this paper is structured as follows. The needed background and an overview of related work are presented in Section II. A scenario is provided in Section III to illustrate how to analysis distance between switch and controller. Also, the proposed model is presented regarding the new definition of latency. Evaluation and results are presented in Section IV. Conclusions and future work are outlined in Section V.

2. Background and Related work

A. Background

1) Facility Location Problem (Mathematically):

The facility location problem (FLP) is the general problem behind the SDN controller placement problem [15]. FLP appears in many contexts such as manufacturing plants, storage facilities, depots, warehouses, libraries, fire stations, hospitals, and base stations for wireless services. Before going into more details, first, a review is presented about location problem and its classification.

Facility location problems deal with selecting the placement of a facility to best meet the demanded constraints. Location problems and models can be classified in a number of ways. The classification may be based on the topography that is used, the number of facilities to be located, the nature of the inputs, whether there is one objective or multiple objectives, whether the facilities are of unlimited capacity or are capacitated, and a variety of other classification criteria [18].

Another way of characterizing facility location problems is by the number of facilities to be located. In some problems (e.g., the P-median, f-center, and maximum covering problems), the number of facilities to locate is exogenously specified. In other cases (e.g., the set covering problem and the fixed charge facility location problem), the number of facilities is endogenous to the problem and is a model output. For those problem statements in which the number of facilities to locate is exogenously specified.

We also distinguish between single-facility location problems and those in which multiple facilities are to be sited. Often, single-facility location problems are dramatically easier than are their multifacility counterparts [18].

In many location contexts, service to customers depends on the distance or time between the customer and the facility to which the customer is assigned. Often, service is considered adequate if the customer is within a given distance of the facility and is considered inadequate if the distance exceeds some critical value. Such problems are called “covering” problems, which require each demand to be served or “covered” within some maximum time or distance standard [18].

A demand is defined as covered if one or more facilities are located within the maximum distance or time standard of that demand [19]. Our SCCPPM model can be related to set covering problem. Similarly, the task of the model is to find the minimal number of controllers such that each switch is no farther than a pre-specified distance away from its closest facility.

B. Related Work

We review main research areas related to the resilience of SDN networks in this section and distinguish our present study on SDN controller placement from related work.

1) Resilience in SDN:

One of the major goals of networking is to achieve resilient communication. Because of the split architecture of SDN and multiple possible failures in different pieces ones (of the architecture), this issue needs to be investigated further. A number of related work have started to tackle the concerns around resilient in SDN.

Jivorasetkul et al. proposed an end-to-end header compression mechanism for reducing latency in SDN networks and thus improving convergence time [9].

Sharma et al. [8] focused on fault tolerance of SDN to deploy it in carrier-grade networks. In order to meet carrier grade requirements, they utilized two recovery mechanisms (restoration and protection) and added the recovery action in the switches themselves. Besides, the delegated some control functions from the controller to switches so that the switches can do recovery without contacting the controller. They demonstrate that such approach can achieve recovery in the order of 100 ms in a large-scale network.

Another related line of work is SlickFlow [20], leveraging the idea of using packet header space to carry alternative path information to implement resilient source routing in Open-Flow networks. Under the presence of failures along a primary path, packets can be re-routed to alternative paths by the switches themselves without involving the controller.

In [10], a new model based on modifying Open-Flow was proposed. Devo-Flow improves resiliency by delegating some work to the forwarding devices. For instance, instead of requesting a decision from the controller for every flow, switches can selectively identify the flows (e.g., elephant flows) that may need higher-level decisions from the control plane applications.

DIFANE is another scalable and efficient solution that reduces a load of a centralized controller by distributing network state among switches [21]. This scheme keeps all traffic in the data plane by selectively directing packets through intermediate switches that store the necessary rules. DIFANE relegates the controller to the simpler task of partitioning these rules over the switches.

Furthermore, several efforts have been made to tackling performance and distributed control plane, including Maestro [11], SDX [12], Onix [14], and NOX-MT [22]. Overall, one cannot say that distributed control plane and high-performance controllers cause high resiliency and quickly respond to network events.

2) Controller Placement Problem:

The controller placement problem has been discussed in a couple of papers. This problem is comparable with facility location problem in many aspects. Heller et al. established the first study in this area [15]. Using a brute-force method, they evaluated the impact of controller placement on average and maximum latency metrics for real network topologies. They show that in most topologies one single controller is enough to fulfill ‘existing reaction-time requirements’.

In [23], a mathematical model was presented for the capacitated controller placement that predicts failures to prevent a significant growth in worst-case latency and disconnections. Indeed, if there are multiple controllers, reallocating switches of the failed controller may considerably raise the worst-case latency. The model aims at minimizing the worst-case latency between switches and their Kth reference controllers such that the capacity and closest assignment constraints are satisfied.

Kshira and et al. proposed two population-based meta-heuristic algorithms, Firefly and Particle Swarm Optimization (PSO), for optimal placement of the controllers. These algorithms take a particular set of objective functions and return the best possible position in comparison with previous works [24].

In [25], authors presented a dynamic controller placement model that consists of determining the locations of controller modules to optimize latencies, and the number of controllers per module to support the load. The provisioning of controllers at each of these controller modules is to handle the dynamic load.

In [26], authors propose an energy-aware traffic engineering solution, called GreCo. They proposed a controller association algorithm to address the assignment of switches to controllers under an energy saving consideration, where they assumed that the controller placement was already known.

As can be seen, most work on the topic of controller placement in literature concentrates on the fact that the problem is NP-hard and depending on some objectives often provide only approaches to the location of controllers; while determining the number of required controllers regarding some requirements is a prerequisite input for the former.

3. Problem Definition

A. Analysis of Latency (Restoration Time)

In order to calculate bidirectional end-to-end delay between switches and their controllers (or restoration time), we first need to assess imposed latencies in network devices whenever an event occurs. Following scenario explains serial steps to handle an event regarding their latencies.

1. When a switch detects an event, it performs required processes and sends a request to the controller to get instructions on how to handle the event or any other variation. Here, processing delay in the switch includes the required time to process the event and generate a notification message.

2. The notification message is sent to the controller through allocated links between the switch and its controller. This step consists of multiple delays: processing delay and queuing delay in intermediate switches, transmission delay, and propagation delay.
3. Then, the controller will decide how to process/handle that event. It performs required processes at the behest of the switch. It sends requiring instructions back to the switch, through processing delay and queuing delay in the controller.
4. Required rules are sent back to the switch. Similar to Step 2, we have multiple delays: processing delay and queuing delay in intermediate switches, transmission delay, and propagation delay.
5. The switch updates its entries or installs new rules in the flow table to conduct the incident occurred. In this way, it processes the delay and queue delay in the switch including the required time for updating the switch as the flow changes.

As can be seen, restoration time is the sum of the processing times and queuing times in the switch and the controller and intermediate switches and the time for transmitting and propagating messages through links in both directions.

Let d_{Sproc}, d_{prop}, d_{tran}, $d_{interSproc}$, $d_{interSque}$, d_{Cproc}, d_{Cque}, and $d_{Supdate}$ denote the processing delay in the intended switch, propagation delay, transmission delay, processing and queuing delays in intermediate switches, processing and queuing delays in the controller and flow table update delay in the intended switch, respectively. Then the total end-to-end bidirectional delay between switch and controller is given by:

$$\mathrm{D_{C2N} = D_{Sproc} + 2D_{prop} + 2D_{tran} + 2D_{InterSproc} + 2D_{InterSque} + D_{Cproc} + D_{Cque} + D_{Supdate}} \quad (1)$$

As can be seen, the restoration time in SDN is totally different compared to that in traditional networks. Because returning to a previous state after the occurrence of an event in an Open-Flow switch requires instructions from the Open-Flow controller, it usually results in longer restoration time compared with legacy network. Indeed, many processes exist that affect the restoration time or bidirectional end-to-end delay[1] in SDN.

Each of these latency metrics depends on various factors. For instance, the processing and queuing latencies in intermediate switches are widely varied depending on network load and the hardware switch platform. The time for flow table update and processing delay, $d_{Supdate}$ and d_{Sproc} respectively, in the intended switch can be quite high and varying. They depend on many factors such as flow table size, the switch platform, flow setup rate, event rate, rule priority, and a load of the network [27] [28]. However, transmission delay can be negligible if notification messages and forwarding rules are small. Propagation delay can be determined by topology graph.

As can be seen, the total end-to-end bidirectional delay between switch and controller is highly varying, depending on many factors. Few researchers have investigated some of these delay metrics separately [27] [28]. They carried out various setup experiments to measure convergence time. They changed the protection schemes, number of switches, number of threads, occupancy of flow table, hardware platform switch, controller workload, and many other parameters. Based on their experiments and simulations, they concluded that the restoration time varies between 10 and 40 ms or even more [8] [29] [30]. It is important to note that most of the experiments are done in out of band control plane.

The calculation of each one of these latencies is outside the scope of this article and is a direction for future work. It can be modeled by queuing theory, calculus network, and other mathematical models. Here, we optimistically consider this delay within the range of 15 to 25 ms randomly. Therefore, the total end-to-end bidirectional delay between switch and controller is given by:

$$\mathrm{D_{C2N(end-to-end)} = 2D_{prop} + D_{variable}} \quad (2)$$

Propagation latency between the switch and the controller is denoted by d_{prop}, which is extracted from internet topology ZOO and $d_{variable}$ indicates other mentioned latencies. In the evaluation phase, we optimistically consider this delay is between 15 to 25ms randomly in our model.

B. Proposed Model (SCCPP Model)

So far, we have been familiar with the concept of resiliency and the calculation of latency between each switch and its assigned controller in SDN. The aim of this study is to find the minimum number of controllers required in the specific topology so that the maximum amount of the latency becomes less than or equal to 50 milliseconds. For this purpose, we model this problem as an Set Covering Controller Placement Problem (SCCPP) regarding latency constraint. A network is given. Let:

Data

G : the network that the decision maker locates controllers;

n : the number of nodes (or switches);

V : set of vertices in the network;

E : the set of physical links between the nodes, and the weight of each edge between two nodes represent the propagation latencies;

D_{ij} : the shortest path from node $i \in V$ to $j \in V$ according to the propagation latency;

D_{proc} : the time required it takes controllers and switches to process needed;

D_c : maximum coverage distance (let $D_c = 50$ milliseconds be coverage distance);

f_i : cost of locating a controller at candidate site i;

N_j : the set of controllers eligible to provide "cover" to switch j:

$$\mathrm{N_j = \{i \in V \middle| D_{ij} + D_{proc} \leq D_c \}}$$

Decision Variables

[1] In some paper this called (well known as) flow setup time or convergence time.

$$\mathrm{Z_i} = \begin{cases} 1 & \text{if a controller is located at site i,} \\ 0 & \text{otherwise,} \end{cases}$$

$$\mathrm{X_{ij}} = \begin{cases} 1 & \text{if switch j is assigned to controller i,} \\ 0 & \text{otherwise,} \end{cases}$$

With this notation, we can formulate the problem as follows:

$$\text{minimize} \sum\nolimits_{\mathrm{i}} \mathrm{f_i Z_i} \qquad (3)$$

Subject to:

$$\sum\nolimits_{\mathrm{i \in N_j}} \mathrm{X_{ij}} = 1 \quad \forall \mathrm{j} \qquad (3-\mathrm{a})$$

$$\sum\nolimits_{\mathrm{j}} \mathrm{X_{ij}} \leq \mathrm{nZ_i} \quad \forall \mathrm{i} \qquad (3-\mathrm{b})$$

$$\mathrm{Z_i} \in \{0, 1\} \qquad (3-\mathrm{c})$$

$$\mathrm{X_{ij}} \in \{0, 1\} \qquad (3-\mathrm{d})$$

The objective function (3) minimizes the total cost of the selected controllers. Constraints (3-a) state that each switch *j* must be covered by exactly one controller within the maximum time or distance standard D_c. Constraints (3-b) require that if site (node) *i* is selected to establish a controller, its maximum assigned switches would be *n*. Otherwise, no switches could be dedicated to this, as it is not a controller. Constraints (3-c) and (3-d) are the integrality constraints. Cost of all coefficients f_i can be measured in the terms more related to the network operator; for instance, they could be considered as economic cost. If all the cost coefficients f_i are equal, the problem is called the uni-cost SCP and the objective function may be simplified as follow:

$$\text{minimize} \quad \sum\nolimits_{\mathrm{i}} \mathrm{Z_i} \qquad (4)$$

Before proceeding further, we should note that since this problem is a set covering problem, it proves that the problem on a general graph is NP-complete. This is true for either objective function (1) or objective function (2) [18].

4. Evaluation and Result

A. Analysis of NTT topology - case study

Our interest in developing models (3) and (4) lies in their ability to analyze existing network topologies from the perspective of deploying carrier-grade SDNs with regard latency. Given a topology, if it will be implemented based on SDN architecture, there is a set of candidate sites where controllers can be deployed. At the outset, we need to know how many controllers are required.

For this purpose, first, we examine the proposed model for NTT communication network and analyze it, followed by investigating further a larger number of topologies from internet topology ZOO in the next section. NTT is one of the largest carrier-grade infrastructure providers in the world, with its services reaching 160 countries/regions including the most extensive coverage in the Asia Pacific.

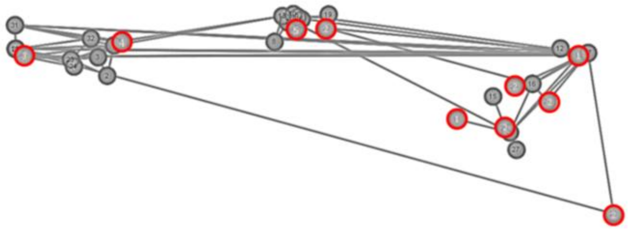


Fig. 1. The minimum number of required controllers in NTT network topology

Moreover, it has been a pioneer in the arena of software-defined networking. NTT is going to take the advantage of architectures based on SDN for delivering its services, and it has been deploying SDN/Open-Flow to connect 17 of its global data centers for almost three years [31].

The network topology related to the infrastructure of this carrier is presented in Figure 1. This topology contains 32 nodes and it has spread throughout the world. Considering the characteristics of this topology, which is stretched in large scale, controller placement problem is critical for it. Model (3) was applied to this topology to find out the minimum number of the required controllers.

To provide some intuition for placement considerations, Figure 1 shows the minimum number of controllers and their placements. By solving this model in CPLEX 12.6, it can be found that to achieve carrier-grade requirements, the required controllers must be at least 10 controllers with their locations being at 4, 5, 11, 18, 20, 21, 22,2 5, 28, and 30.

B. Analysis of more Topologies - Case Study

In this section, we expand our analysis to 124 topologies from Internet Topology Zoo with graph sizes ranging from 25 to 65 nodes randomly. This dataset includes a collection of network graphs derived from public network maps covering a diverse range of geographic areas, network sizes, and topologies. The proposed model, solved through CPLEX 12.6, is applied to these topologies. The obtained results are presented in Figure 2 with a confidence interval of α=0.05. The eight values written on the horizontal axis represent bins' thresholds. The first bin contains all topologies for which the network size contains up to 30 nodes, the second bin comprises of all topologies for which $30 < number\ of\ nodes \leq 35$, and so on. The vertical axis illustrates the minimum number of required controllers regarding the latency constraint. The graph shows that the number of required controllers for a high resiliency varies highly. Besides, it strongly depends on the network topology than network size.

Figure 3 presents the cumulative distribution of minimal required controllers for each topology. From these results, we can see that to achieve carrier-grade requirement, 86% of topologies must have more than one controller.

Furthermore, one controller is only enough for 14% of topologies, contrary to the results of Heller et al. reported [15] who reported that "one controller location is often sufficient to meet existing reaction-time requirements".

Nonetheless, 5 controllers are enough for 95% of scenarios even looking for a high resiliency.

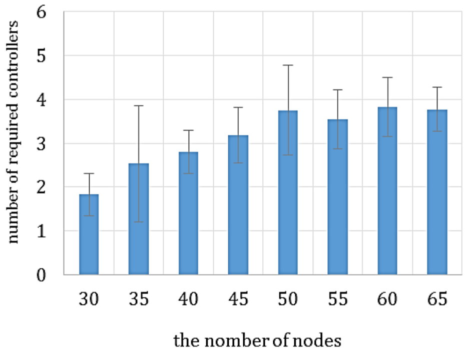


Fig. 2. Size of topology and the corresponding minimum number of required controllers

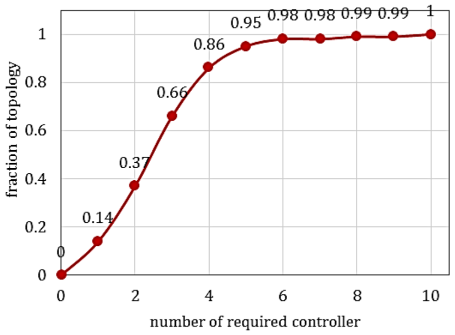


Fig. 3. The CDF of minimal required controllers for each topology

5. Conclusion and Future Work

Achieving a high resilient communication is one of the most important goals in networking. In this paper, we initiated the study of the resilience in SDN from the controller placement standpoint. Furthermore, we focus more on two important issues in CPP-related works that have received less attention. First, we exhausted the subject of latency between the switch and its controller and found this metric is highly varying and it depends on various conditions. For future works, this scheme can be extended by other network modeling such that to estimate the distance between switch and controller more accurately. Moreover, such approach should be followed to calculate the latency between controllers especially when in-band control is in play. Secondly, a SCCPPM is proposed to find the minimal number of controllers with regard to latency constraint. Normally, it is essential to determine the number of required controllers with respect to administration requirement followed by finding their optimal location. Indeed, the former is a prerequisite act before applying the latter. Finally, we conclude that 86% of topologies must have more than one controller to achieve carrier grade requirement. However, we absolutely need more than this, if we consider some other administration constraints.

References

[1] Kreutz, D., Ramos, F. M., Verissimo, P. E., Rothenberg, C. E., Azodolmolky, S., & Uhlig, S. (2015). Software-defined networking: A comprehensive survey. Proceedings of the IEEE, 103(1), 14-76.

[2] Jain, S., Kumar, A., Mandal, S., Ong, J., Poutievski, L., Singh, A., and Zolla, J. (2013). B4: Experience with a globally-deployed software defined WAN. ACM SIGCOMM Computer Communication Review, 43(4), 3-14.

[3] Sezer, S., Scott-Hayward, S., Chouhan, P. K., Fraser, B., Lake, D., Finnegan, J., and Rao, N. (2013). Are we ready for SDN? Implementation challenges for software-defined networks. IEEE Communications Magazine, 51(7), 36-43.

[4] Kano, S., Miyazaki, K., Nagata, A., and Chugo, A. (2005, November). Shared segment recovery mechanism in optical networks. In 6th Asia-Pacific Symposium on Information and Telecommunication Technologies (pp. 415-420). IEEE.

[5] Laprie, J. (2005, July). Resilience for the scalability of dependability. In Fourth IEEE International Symposium on Network Computing and Applications (pp. 5-6). IEEE.

[6] Benzekki, K., El Fergougui, A., and Elbelrhiti Elalaoui, A. (2017). Software- defined networking (SDN): a survey. Security and Communication Networks.

[7] Oliveira, D., Pourvali, M., Bai, H., Ghani, N., Lehman, T., Yang, X., and Hayat, M. (2017, March). A novel automated SDN architecture and orchestration framework for resilient large-scale networks. In SoutheastCon, 2017 (pp. 1-6). IEEE.

[8] Sharma, S., Staessens, D., Colle, D., Pickavet, M., and Demeester, P. (2013). OpenFlow: Meeting carrier-grade recovery requirements. Computer Communications, 36(6), 656-665.

[9] Jivorasetkul, S., Shimamura, M., and Iida, K. (2013, August). Better network latency with end-to-end header compression in SDN architecture. In Communications, Computers and Signal Processing (PACRIM), 2013 IEEE Pacific Rim Conference on (pp. 183-188). IEEE.

[10] Curtis, A. R., Mogul, J. C., Tourrilhes, J., Yalagandula, P., Sharma, P., and Banerjee, S. (2011). DevoFlow: Scaling flow management for high-performance networks. ACM SIGCOMM Computer Communication Review, 41(4), 254-265.

[11] Ng, E., Cai, Z., and Cox, A. L. (2010). Maestro: A system for scalable openflow control. Rice University, Houston, TX, USA, TSEN Maestro-Techn. Rep, TR10-08.

[12] Mambretti, J., Chen, J., Yeh, F., Grossman, R., Nash, P., Heath, A., and Zhang, Z. (2017, March). Designing and deploying a bioinformatics software-defined network exchange (SDX): Architecture, services, capabilities, and foundation technologies. In Innovations in Clouds, Internet and Networks (ICIN), 2017 20th Conference on (pp. 135-142). IEEE.

[13] Canini, M., Salem, I., Schiff, L., Schiller, E. M., and Schmid, S. (2017, June). A self-organizing distributed and in-band SDN control plane. In Distributed Computing Systems (ICDCS), 2017 IEEE 37th International Conference on (pp. 2656-2657). IEEE.

[14] Koponen, T., Casado, M., Gude, N., Stribling, J., Poutievski, L., Zhu, M., and Shenker, S. (2010, October). Onix: A distributed control platform for large-scale production networks. In OSDI (Vol. 10, pp. 1-6).

[15] Heller, B., Sherwood, R., and McKeown, N. (2012, August). The controller placement problem. In Proceedings of the first workshop on Hot topics in software defined networks (pp. 7-12). ACM.

[16] Zhang, Y., Cui, L., Wang, W., and Zhang, Y. (2017). A Survey on Software Defined Networking with Multiple Controllers. Journal of Network and Computer Applications.

[17] McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., and Turner, J. (2008). OpenFlow: enabling innovation in campus networks. ACM SIGCOMM Computer Communication Review, 38(2), 69-74.

[18] Farahani, R. Z., Asgari, N., Heidari, N., Hosseininia, M., and Goh, M. (2012). Covering problems in facility location: A review. Computers & Industrial Engineering, 62(1), 368-407.

[19] Li, S., and Huang, Y. (2014). Heuristic approaches for the flow-based set covering problem with deviation paths. Transportation Research Part E: Logistics and Transportation Review, 72, 144-158.

[20] Ramos, R. M., Martinello, M., and Rothenberg, C. E. (2013, October). Slickflow: Resilient source routing in data center networks unlocked by openflow. In Local Computer Networks (LCN), 2013 IEEE 38th Conference on (pp. 606-613). IEEE.

[21] Yu, M., Rexford, J., Freedman, M. J., and Wang, J. (2010). Scalable flow-based networking with DIFANE. ACM SIGCOMM Computer Communication Review, 40(4), 351-362.

[22] Tootoonchian, A., Gorbunov, S., Ganjali, Y., Casado, M., and Sherwood, R. (2012). On Controller Performance in Software-Defined Networks. Hot-ICE, 12, 1-6.

[23] Killi, B., and Rao, S. Optimal Model for Failure Foresight Capacitated Controller Placement in Software Defined Networks, (2016, June), Communications Letters, IEEE, 20(6), 1108 - 1111.

[24] Sahoo, K. S., Sarkar, A., Mishra, S. K., Sahoo, B., Puthal, D., Obaidat, M. S., and Sadun, B. (2017). Metaheuristic Solutions for Solving Controller Placement Problem in SDN-based WAN Architecture

[25] ul Huque, M. T. I., Si, W., Jourjon, G., and Gramoli, V. (2017). Large-Scale Dynamic Controller Placement. IEEE Transactions on Network and Service Management, 14(1), 63-76.

[26] Ruiz-Rivera, A., Chin, K. W., and Soh, S. (2015). Greco: an energy aware controller association algorithm for software defined networks. IEEE Communications Letters, 19(4), 541-544.

[27] He, K., Khalid, J., Das, S., Gember-Jacobson, A., Prakash, C., Akella, A., and Thottan, M. (2015, June). Latency in software defined networks: Measurements and mitigation techniques. In ACM SIGMETRICS Performance Evaluation Review (Vol. 43, No. 1, pp. 435-436). ACM.

[28] Kuźniar, M., Perešíni, P., and Kostić, D. (2015, March). What you need to know about SDN flow tables. In International Conference on Passive and Active Network Measurement (pp. 347-359). Springer International Publishing.

[29] Sood, K., Yu, S., Xiang, Y., and Cheng, H. (2016). A General QoS Aware Flow-Balancing and Resource Management Scheme in Distributed Software-Defined Networks. IEEE Access, 4, 7176-7185.

[30] Rechia, F. S. (2016). An Evaluation of SDN Based Network Virtualization Techniques (Doctoral dissertation, ARIZONA STATE UNIVERSITY).

[31] P. Bernier, “NTT Recognized with IBC Award for SDN-based HDTV Service,” September 2013. [Online]. Available: http://www.sdnzone.com/topics/software-defined-network/articles/353466-ntt-recognized-with-ibc-award-sdn-based-hdtv.html.

Ahmad Jalili born in 1987. Now he is a Ph.D. candidate on Computer Networks in Shiraz University of Technology. He has many publications in international conferences and journals regarding Wireless Sensor Networks (WSNs) and Software Defined Networks (SDNs). Currently he focused on Software Defined Networks (SDN) as a new trend in computer networks. His major fields of interest are Software Defined Networks (SDNs), heuristic algorithms, Wireless Sensor Networks (UWSNs), Ad hoc Networks and Modeling.

Manijeh Keshtgari received her B.Sc. in Computer engineering from Shiraz University in 1986, her Master in Electrical and Computer Eng. from Colorado State University, USA in 1992 and Ph.D. degree in Computer Eng. from Sharif University of Technology in 2004. Her research interests include wireless Networks, Fiber Optic Networks, Software Defined Networking (SDN) and Named Data Networking (NDN). He is now member of faculty and lecturer in Computer Engineering and IT Department in Shiraz University of Technology. In addition she is lecturer of Computer Science in Department of Computer Science in University of Georgia, USA.

Reza Akbari received his M.Sc. in artificial intelligence from Isfahan University of Technology (2006). He has received his Ph.D. from Shiraz University (2011). He is an Assistant Professor at the department of computer engineering and information technology of the Shiraz University of Technology. His areas of research include evolutionary computation, engineering optimization, and search based software engineering.

Publication Venue Recommendation Based on Paper Title and Co-authors Network

Ramin Safa
Department of Engineering, University of Guilan, Guilan, Iran
r.safa@outlook.com
SeyedAbolghasem Mirroshandel*
Department of Engineering ,University of Guilan, Guilan, Iran
mirroshandel@gmail.com
Soroush Javadi
Department of Engineering , University of Guilan, Guilan, Iran
soroush.javadi@gmail.com
Mohammad Azizi
Department of Engineering , University of Guilan, Guilan, Iran
Moh.Azizi@gmail.com

Received: 13/Aug/2017 Revised: 07/Dec/2017 Accepted: 03/Jun/2018

Abstract

Information overload has always been a remarkable topic in scientific researches, and one of the available approaches in this field is employing recommender systems. With the spread of these systems in various fields, studies show the need for more attention to applying them in scientific applications. Applying recommender systems to scientific domain, such as paper recommendation, expert recommendation, citation recommendation and reviewer recommendation, are new and developing topics. With the significant growth of the number of scientific events and journals, one of the most important issues is choosing the most suitable venue for publishing papers, and the existence of a tool to accelerate this process is necessary for researchers. Despite the importance of these systems in accelerating the publication process and decreasing possible errors, this problem has been less studied in related works. So in this paper, an efficient approach will be suggested for recommending related conferences or journals for a researcher's specific paper. In other words, our system will be able to recommend the most suitable venues for publishing a written paper, by means of social network analysis and content-based filtering, according to the researcher's preferences and the co-authors' publication history. We used the minimum available free features and the minimum implementing facilities, which to the best of our knowledge have not seen up to now. In addition, it can be argued that the proposed system overcome the cold start problem which has always been a remarkable task in recommender systems. The results of evaluation using real-world data show acceptable accuracy in venue recommendations.

Keywords: Academic Recommender Systems; Social Network Analysis; Publication Venue Recommendation; DBLP.

1. Introduction

Recommender systems include software tools and techniques which recommend the most appropriate options by using different kinds of knowledge, user-related data, existing items and previous transactions. They will enable users to achieve their goal more quickly in a large amount of information [1]. Generally, there are three kinds of methods for recommendation: collaborative filtering, content-based filtering, and hybrid systems.

Briefly, in the collaborative filtering, the system identifies the items which may be interesting to a user by taking advantage of previous behaviors and finding similar rating patterns. In the other hand, in content-based filtering, a model of user preferences is created according to the features of the items. In this method, by identifying items similar to which the user liked contently before, and matching user's profile and items' features, the system presents recommendations [1–4]. However, both of these approaches have some issues. For example, data availability and data quality have always been significant subjects in collaborative filtering, and syntactic issues and compound nouns are important topics in content-based filtering. So, in hybrid systems, the quality of recommendations is improved by using the advantages of both aforementioned methods.

Recently, lots of attention have been paid to utilizing the information of users' social network and the existing relations for customization. There is also a growing trend in research on the use of recommender systems in social networks, especially in scientific environments [5]. Scientific social networks are resources that include relations among researcher, publications and bibliographic information which help knowledge development by sharing scientific publications. The large number of scientific bases, papers, and fields of research reveal the importance of using recommender systems and content personalization [6–8].

In online scientific communities, applying recommender systems is observed in fields such as paper recommendation [9–14], expert recommendation [8,15],

* Corresponding Author

reviewer recommendation [16,17] and citation recommendation [18,19]. Due to the rapid growth of the number of the scientific events, especially in computer science [20], few attempts are done towards venue recommendation for publishing a new paper.

Searching for a conference or journal whose scope matches a new paper's topic can be difficult and time-consuming, and will not always lead to desirable results. This can motivate us to employ recommender systems, as the venues suitable for publishing a paper can be extracted by recognizing the researcher's preferences and applying the information filtering process. This could help the researchers, especially those who have no enough experience to choose the most appropriate venues from a lot of conferences and journals.

Despite the fact that some online publishers like Elsevier[1] and Springer[2] try to recommend their related journals by asking user to provide some information about written paper, the comprehensive system with the ability of recommending effective conferences and journals (taking diversity into account) has not been found in practice. In addition, small number of studies conducted in this area are mostly assuming that all of the information for implementing their approach are always available, which regardless of data gathering and matching cost issues, it cannot to be mapped to the real-world and seems to be not applicable. In this paper we present an approach for venues recommendation based on paper's title and co-authors network. As bibliographic information of papers and publication metadata such as title, author(s), year, and venue are freely available, can be used to compute similarity measures and find related documents. We also believe that people who are close to each other have the same taste. As a result, the publications of a co-author can be an informative clue for recommending scientific venues.

The rest of this paper is structured as follows: in the next section, we briefly survey the related work on venue recommendation. In Section 3, we present and detail our approach to recommendation using paper's title and co-authors network. Section 4 will be about another approach—singular value decomposition. In Section 5, we describe the experimental setup and discuss the results in detail. Finally, the paper will be concluded with a summary and future work in Section 6.

2. Related Work

Recommender systems help encounter problems resulted from the explosive growth of information and facilitate decision-making and selection based on user's interest through information filtering process [1]. It is proven that recommender systems are useful and valuable tools to encounter the problems resulted from information overload for online users. Nowadays, recommender systems are used in different fields—e.g., e-commerce, news, and entertainment—but researches show a high demand for utilizing these systems in scientific domains [6,21].

Social relations of people can influence their behavior and interests, hence we observe utilizing social network capabilities in different domains, and social network analysis in recommender systems is a new emerging topic [8,21]. By combining traditional methods of recommender systems and social network analysis, more effective recommendations can be achieved. Social interactions in scientific communities—such as co-authorship and participation in similar conferences—can influence recommendation quality.

Klamma et al [20] recommended conference venues to researchers by utilizing collaborative filtering concepts. Their system uses DBLP[3] and Eventseer.net information about venues. It extracts some useful information about individuals who participated in similar conferences to those the user participated in, to recommend related scientific events. The content-based approach is not employed in this research. Also, recommendations presented to a user are general, and not specific to a written paper.

It should be noted that in scientific communities, the semantic relations between papers and their publication venues are considered important, and collaborative filtering will not be able to extract these relations. This approach only takes interactions between users and items into account [12].

Martı́n et al [22] proposed a content-based filtering algorithm. Their algorithm uses textual information of call for papers and recent paper abstracts of each conference program committee member, and also the abstracts of the user's recent papers and their citation information.

Medvet et al [23] suggested a system which used paper title and abstract for recommending a publication venue. They extract conference papers in computer science from Microsoft Academic Search[4] engine, which most have the necessary features, and by matching the title and abstract of the user's new paper with conference selected papers try to recommend appropriate venue to the user.

Xu et al [24] studied a comprehensive system that covers all aspects of paper life cycle. In this work, conference and journal recommendation is mentioned as the most important part of the system. Their proposed system extracts the keywords from context, and then, recommendation process will be a subject-oriented query. It should be noted that their system does not utilize co-authors network and social network analysis.

To the best of our knowledge, utilizing authors social network for publication venue recommendation has been introduced by Luong et al [7] for the first time. Three methods are presented in their research to recommend relevant publication venues using social network analysis: (1) most frequent conference, (2) most frequent conference normalized by author, and (3) the second

[1] Available from: http://journalfinder.elsevier.com
[2] Available from: https://journalsuggester.springer.com
[3] Available from: http://dblp.uni-trier.de/db
[4] Available from: http://academic.research.microsoft.com

method combined with network topology. In the first method, using co-authors social network information, the conference in which these people had published the most number of papers will be recommended. The second method utilizes normalization in order to decrease the influence of the authors who wrote the most number of papers on the final results. In the third method, the weight of authors that have more previous collaborations with the main author will be considered more important. Results show the superiority of the network-based approach compared to the content-based one. It should be noted that paper content is not used in their work.

Beierle et al [25] identified six main ways for extracting relation of two authors in academic social graph, through common publications (co-authorship), affiliations, similar keywords (co-interests), commonly visited venues (co-activity), referencing or being referenced by the other. Based on their research, social relations can be used to derive author's preferences and exploited for conference recommendations. Furthermore, they found co-authorship, co-interests, and co-activity lead to the best recommendation accuracy. This is almost the same point mentioned by García et al [26].

We can also take a brief look at Peiris and Weerasinghe [27] work, who proposed an approach for ranking publication venues by considering publication history and citation network. They believe there are some aspects that contribute the importance of a publication, including citation it has received, the quality of the citing publications, the time metric and its authors.

In this paper, we present an approach to recommend related venues for a user's certain paper, by employing social network analysis concepts and content-based filtering. Experimental results using real-world data show that our approach can provide effective recommendations.

3. Proposed Method

Scientific recommendations are often done using content-based filtering [13] and based on paper content. It is necessary to mention that obtaining the full-text version of papers is usually not possible due to copyright issues, but bibliographic information of papers and publication metadata such as title, author(s), year, and venue, which can be a useful source of information, are freely available, and can be used to compute similarity measures and find related documents.

Generally, deriving a user's interests can be done in two ways: explicit and implicit. In the former, the user declares his/her preferences explicitly, while in the latter, the user's preferences are identified by monitoring and analyzing his/her activities [1,28]. In our studies, it is observed that some paper retrieval systems [13,29,30] and deriving users' profile algorithms [31] have used words appearing in a user's publication title as his/her interests.

On the other hand, it is proven in different researches that people tend to accept recommendations that are presented by people around them, rather than those who are far from them but have similar tastes. This opens many research opportunities subsumed under social recommender systems [1,28,32]. In scientific domain, studying models and methods presented in various researches, it can be concluded that utilizing the information about people around a user increases recommendation accuracy [7,14,15,33].

The main idea is that we can use similarity measures with co-authors' published papers and the target paper, to recommend related conferences and journals for publishing the paper.

As shown in figure 1, our recommender system takes author(s) identity and the new paper's title as input. It finds similar papers of co-authors by matching their titles with the input title. The venues of extracted papers will be ranked and recommended. This procedure can be done recursively for co-authors of co-authors and so on.

Figure 1 shows the architecture of our system which mainly consists of three components: linguistic information extractor, matcher, and ranker.

3.1 Linguistic Information Extractor

Documents should be changed into a structured form to be interpretable for the system in order to apply our similarity measure. In natural language processing, there is a procedure called stop word removal, which is done to remove the most frequent used words. In this component, a definite list of stop words taken from MySQL website[1] is removed from each paper's title. It is necessary to mention that to enhance the results, words "a", "i" and "based" are added to this list.

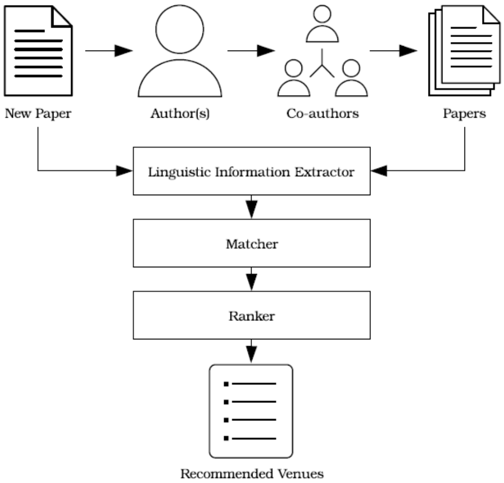


Fig. 1. Proposed method overview

In language morphology and information retrieval, there is another process named stemming, that aims towards reducing a word to its root, by removing some parts of the word (the affixes). Applying stemming is very important in matching process [34].

After removing stop words, stemming is done by the Porter2 stemmer[2], which is the improved version of the Porter stemmer with minor modifications. Porter is one of

[1] Available from: http://bit.ly/mysqlstopwords
[2] Available from: http://bit.ly/porter2stemmer

the most well-known tools for stemming and it is commonly used for the English language due to its high performance. Some of Porter's rules are as follows [35]:

- sses → ss
- ies → i
- s → *(null)*
- izer, ization → ize
- ator, ation, ational, → ate

In other words, after taking papers' titles, stop word removal is applied to remove uninformative words, and then stemming is done by the Porter2 stemmer in order to increase matching accuracy by reducing the diversity of words.

3.2 Matcher

In order to apply matching and similarity measures, each document is represented by a vector of term weights (i.e., term frequencies). Next, cosine similarity is used, which is one of the most appropriate similarity measures. This measure is defined as follows, where *A ·B* denotes the dot product of the vectors *A* and *B* [1]:

$$\mathrm{similarity(A, B)} = \cos\theta = \frac{\mathrm{A{\cdot}B}}{\|\mathrm{A}\|\|\mathrm{B}\|} = \frac{\sum_{i=1}^{n} \mathrm{A_i}\times\mathrm{B_i}}{\sqrt{\sum_{i=1}^{n}(\mathrm{A_i})^2}\times\sqrt{\sum_{i=1}^{n}(\mathrm{B_i})^2}} \qquad (1)$$

3.3 Ranker

After identifying the venues of papers similar to the input paper, weighting is done in order to identify the most relevant venues at ranker component as follows, where P_i is the i^{th} paper published in the venue *V*, and *R* is the new paper:

$$\mathrm{weight(V)} = \sum_{i=1}^{n} \mathrm{similarity(P_i, R)} \qquad (2)$$

Finally, venues are sorted based on their weights and presented to the user.

4. SVD Approach

Singular value decomposition (SVD) is one of the dimensionality reduction techniques which factors an *item × features* matrix *A* into three different matrices: an *item × concepts*, a *concept strength*, and a *concept × features* as the following:

$$\mathrm{A} = \mathrm{U}\lambda\mathrm{V}^{\mathrm{T}} \qquad (3)$$

The most well-known application of SVD in natural language processing is latent semantic analysis (LSA), which is a theory and method for extracting and representing the meaning of words by statistical computations applied to a large corpus of text [36]. LSA can work with a term-document matrix which describes the occurrence of terms in documents. Since it can be used as a prediction tool [37], we considered applying it to our problem as an alternative method.

We utilized SVD to capture latent relationships between terms and venues which allow us to compute our proposed matching algorithm in a different space. To achieve this goal, we started with a term-venue matrix wherein each column represents one of the venues to be ranked, and terms are the words of the preprocessed titles of all published papers in those venues in recent years. Each matrix entry indicates the frequency of the corresponding term in the corresponding venue. Table 1 is an example of term-document matrix for ten venues and a limited number of their terms.

Table 1. An example of term-document matrix

Venue Keyword	CAMAD	EUNICE	HAISA	HPCC-ICESS	IJESMA	ISCA	KMIS	NMR	SPRINGL	SSV
algorithm	2	8	0	24	0	5	0	2	1	1
cellular	2	1	0	1	0	0	0	0	0	0
game	1	1	0	1	0	0	1	1	0	0
hardwar	1	0	1	4	0	18	0	0	1	0
internet	2	6	2	0	2	0	0	0	0	0
mobil	10	8	0	6	17	5	2	0	2	0
network	58	60	4	38	2	25	12	0	3	0
search	0	1	0	1	2	4	1	0	0	0
secur	4	4	29	5	1	12	3	0	4	0
web	0	2	0	3	3	1	13	0	2	0

We computed the singular value decomposition of the term-venue matrix and extracted the singular values, left singular vectors and right singular values from the SVD matrix. Next, we considered the search component of latent semantic indexing. Queries are computed by taking the centroid of the term vectors corresponding to the terms in the query, where the query is the preprocessed title of the new paper. The centroid is computed pointwise, by adding the values in each dimension. This is then matched against the venue vectors using the scaling provided by the singular values, resulting in a score (weight) for each venue. We used dot product and cosine similarity methods for matching.

5. Evaluation and Discussion

The common way to measure the performance of the proposed method is to compare the results of applying it to a standardized dataset with the results of other researchers. But as noted earlier, the number of tasks performed in this field is not high and the datasets of the related works are not available. In the other hand, we received some dataset from the related works, but they do not have all the features and metadata we intended to use and were customized for specific work. Due to different datasets, the comparison of existing methods with our method is not a valid and meaningful comparison. Therefore, we decided to use real-world data and a strict approach to measure the method performance.

We used DBLP data to evaluate the recommendation method. DBLP is the computer science bibliographic website that provides bibliographic information about papers, events and computer science journals. The dataset of this website is saved periodically in an XML file[1] and the last update of this file at the time of evaluation (May 29, 2014) is used to measure the accuracy of the system. This dataset contains eight types of entities which are shown in table 2. As it is described in the table, conference and journal papers are the two major types in this dataset.

Table 2. Types of DBLP entities

Type	Description	Count
Article	Journal Article	1,131,735
Book	Book	10,932
In Collection	Publication Cited in a Collection	26,739
In Proceedings	Publication Published in Conference Proceedings	1,431,399
Master's Thesis	Master's Thesis	9
Ph.D. Thesis	Ph.D. Thesis	6,937
Proceedings	Conference Proceedings	23,146
WWW	Author Links	1,412,090

Each entity contains some of these metadata, among which *title* is the only one that has to exist [38]: *author*, *editor*, *title*, *booktitle*, *pages*, *year*, *address*, *journal*, *volume*, *number*, *month*, *url*, *ee* (electronic edition), *cdrom*, *cite*, *publisher*, *note*, *crossref*, *isbn*, *doi*, *series*, *school* and *chapter*.

In order to evaluate suggested method, we selected a random sample of 20,000 papers among a total of 205,880 papers published in 2013. We used the papers published from 2008 to 2012 as recent papers. Also, to obtain the co-authors of an author, we used the metadata of papers published from 2003 to 2012.

It is necessary to mention that there were some challenges during the preparation of the dataset. In many cases, we observed that the same venue has different names. This is sometimes due to typographical errors, sometimes part of the name is removed or added, and sometimes it is displaced.

Some examples of typographical errors:

- Internet Measurement Conference,
- Internet Measurement Comference, and
- Internet Measurment Conference;
- Computer Supported Activity Coordination, and
- Computer Supported Acitivity Coordination;
- Adaptive Agents and Multi-Agent Systems, and
- Adaptive Agents and Multi-Agents Systems.

Some examples of removal or addition of part of conference name:

- IEEE VAST, and
- VAST;
- GI Jahrestagung, and
- GI-Jahrestagung;
- ICWS, and
- ICWS-Europe.

An example of displacement:

- KR4HC/ProHealth, and
- ProHealth/KR4HC.

Moreover, for venues in whose names there was an at-sign character, we used the part after the at-sign character as the venue name. For example, DUBMOD@CIKM papers were considered as CIKM papers. Also, in a few cases, the paper title contained LaTeX codes, which we ignored.

Among the total 5,865 distinct venues, 611 cases had the aforementioned problems. Venue names containing an at-sign where trimmed automatically. For other cases, we used regular expression search to find them, and corrected them manually. In addition, 164 documents were removed after linguistic preprocessing phase, because no word remained in their title.

After applying the proposed method with depth 1 traversal for co-authors, we observed that in 70.69% of cases, the accurate venue—the venue in which the input paper was actually published—does exist in the system output, and for the top-20 recommendations, accuracy reaches to 48.53%. This seems interesting, because the evaluation was done on a completely random set of DBLP data, and we did not select specific papers with predetermined domains. On the other hand, the average number of total venues reaching the ranker component of our system was 350 venues for each input paper, and the top-20 list of recommendations contains only less than 6% of these possible recommendations. We used the term "Oracle score" in our results to show accountability of the system, which refers to existence of exact venue in total recommendation list of each depth.

For measuring the quality of our recommendations, we could ask human experts to evaluate the output of our system. However, evaluation by human experts is very time-consuming, expensive, and human-dependent. For resolving these issues, we have utilized accuracy measure for evaluation. In our recommender system, accuracy is even stricter measure than evaluation by human experts. In accuracy measure, if the recommended venue is exactly equal to the real published venue, system achieve a score, otherwise the answer will be regarded as unrelated. However, in our observation, lots of recommendation were totally related to the paper, but accuracy did not score these recommendations. We should again emphasis that the satisfaction of researchers (human experts) is the best way for evaluation of the system. In this situation that it is not feasible to evaluate using the mentioned measure, we used accuracy measure.

Co-authors network can be represented as a graph containing authors as nodes and co-author relationship as edges. In the process of evaluation, we learned that the first recommendation in depth 1 of the graph, with an accuracy of 15.18%, has the best accuracy in a recommendation list. The results of the evaluation for recommendation number 1 to 20 can be seen in table 3.

Table 3. Evaluation results

Recommendation no.	**No. of accurate recommendations**	**Accuracy (%)**
Only 1st	3,036	15.18
Only 5th	470	2.35
Only 10th	256	1.28
Only 15th	132	0.66
Only 20th	119	0.59

Table 4 shows system oracle score for depth 1 to 4, that depth 4 has a significant accountability of 88.14%.

[1] Available from: http://dblp.uni-trier.de/xml

Table 4. The system oracle score for depths 1 to 4

Depth	Oracle count	Oracle score (%)
1	14,139	70.69
2	16,361	81.81
3	17,323	86.61
4	17,627	88.14

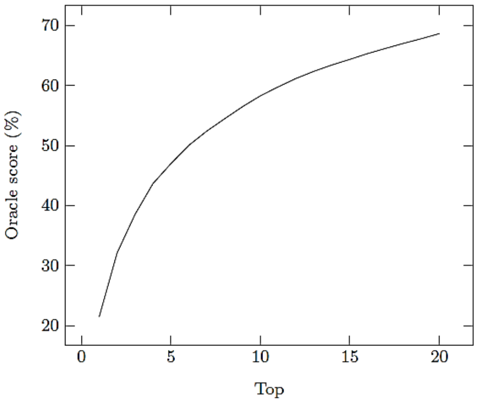


Fig. 2. The oracle score in different tops for depth 1

Tables 5 and 6 show the number of accurate recommendations, the accuracy and oracle score for depth 1 and 2 traversals in different tops.

Table 5. Details of depth 1 evaluation

Recommendations	No. of accurate recommendations	Accuracy (%)	Oracle score (%)
Top-1	3,036	15.18	21.47
Top-2	4,540	22.70	32.11
Top-3	5,447	27.23	38.52
Top-4	6,171	30.86	43.65
Top-5	6,641	33.20	46.97
Top-6	7,073	35.37	50.02
Top-7	7,408	37.04	52.39
Top-8	7,699	38.49	54.45
Top-9	7,984	39.92	56.47
Top-10	8,240	41.20	58.28
Top-11	8,452	42.26	59.78
Top-12	8,652	43.26	61.19
Top-13	8,820	44.10	62.38
Top-14	8,968	44.84	63.43
Top-15	9,100	45.50	64.36
Top-16	9,237	46.19	65.33
Top-17	9,358	46.79	66.19
Top-18	9,478	47.39	67.03
Top-19	9,587	47.94	67.81
Top-20	9,706	48.53	68.65

Evaluation results show that the recommendation accuracy decreases as the traversal depth for co-authors increases. The reason is that, as the depth increases, co-authors grow exponentially in number, but become less relevant. So, even though co-authors of co-authors are obviously less relevant than the co-authors, they are much greater in quantity, hence affect the system accuracy negatively. On the other hand, this decrease in accuracy is accompanied by an increase in the oracle score of the system. It seems that the recommendation accuracy can be enhanced by assigning a proper weight to each depth in order to utilize this accountability in the future.

Table 6. Details of depth 2 evaluation

Recommendations	No. of accurate recommendations	Accuracy (%)	Oracle score (%)
Top-1	2,611	13.05	15.96
Top-2	3,878	19.39	23.70
Top-3	4,696	23.48	28.70
Top-4	5,313	26.57	32.47
Top-5	5,836	29.18	35.67
Top-6	6,296	31.48	38.48
Top-7	6,670	33.35	40.77
Top-8	6,967	34.84	42.58
Top-9	7,235	36.17	44.22
Top-10	7,526	37.63	46.00
Top-11	7,770	38.85	47.49
Top-12	8,002	40.01	48.91
Top-13	8,203	41.02	50.14
Top-14	8,396	41.98	51.32
Top-15	8,558	42.79	52.31
Top-16	8,734	43.67	53.38
Top-17	8,899	44.49	54.39
Top-18	9,040	45.20	55.25
Top-19	9,175	45.88	56.08
Top-20	9,287	46.44	56.76

However, we use the information of author's own network, one may ask about the percentage of new venues in our proposed search space and finally in our recommendations. The answer of this question is shown in table 7. As it is visible in this table, the average number of extracted venues from depth one and two of co-authors network are 349.83 and 1,619.89, respectively. These large numbers show the comprehensiveness of search space. Another interesting column in table 6 is the average percentage of new venues, for the target author, in our top-20 recommendation. Again, this percentage is quite high (i.e., about 80%) and it shows our algorithm is able to discover new venues for recommendation to the user.

Table 7. average number of extracted venues and the average percentage of unfamiliar venues in top-20

Depth	Average no. of extracted venues	Average percentage of new venues in top-20
1	349.83	78.96%
2	1,619.89	83.25%

5.1 SVD Results

Tests of this approach are done on the same dataset. In all tests, cooperation and recent papers are extracted respectively from ten and five recent years. For calculating the SVD, the number of factors (latent semantic dimensions) is restricted in each test.

First, a test in depth 1 is applied to the dataset, with 20 factors and both dot product and cosine and methods. The accuracy of top-20 recommendations for the dot product method was 19.75%, and for the cosine method, was 20.45%. So the cosine similarity method was slightly more accurate.

Increasing the number of factors to 60 in the cosine method improves the accuracy of top-20 recommendations to 22.34%, which is 1.89% better than the same test with 20 factors. The result of this test is shown in table 8.

Table 8. SVD test results in depth 1, with 60 factors and the cosine similarity method

Recommendations	Accuracy (%)
Top-1	4.28
Top-2	6.76
Top-3	8.66
Top-4	10.28
Top-5	11.60
Top-6	12.86
Top-7	13.84
Top-8	14.76
Top-9	15.58
Top-10	16.38
Top-11	17.05
Top-12	17.80
Top-13	18.53
Top-14	19.15
Top-15	19.79
Top-16	20.34
Top-17	20.84
Top-18	21.39
Top-19	21.91
Top-20	22.34

In depth 2, with 20 factors and the cosine method, the accuracy of top-20 recommendations was 10.58%, which is 9.87% less than that of the same test in depth 1.

6. Conclusion and Future Work

The goal of this paper is to design and apply an algorithm to recommend appropriate venues to researchers for publishing scientific papers, by utilizing the title of papers and the co-authors network. We evaluated the proposed method with real-world data from the DBLP computer science bibliography website, and some explanations are presented about the preparation of the dataset and the challenges we encountered. The results of the evaluation show that our method is able to present effective recommendations only by publication metadata and minimum implementing facilities: with depth 1 traversal of the co-authors network, the oracle score of the system was 70.69% and the accuracy was 48.53% for the top-20 recommendations. In this evaluation, it was cleared that system responsibility increases as the traversal depth increases, but with weight tuning for each depth, the accuracy reduction should be prevented.

Our system depends on previous publications of researchers to provide them with venue recommendations, but since novice researchers usually get help from experts in the field of research, it can be argued that the proposed system resolves the cold start problem. Moreover, the combination of the suggested approach with data mining techniques and machine learning algorithms to create a model to find proper venues can be an interesting topic.

To test another approach, we used SVD, and by applying this method to the same dataset with 60 factors, we found out that our suggested system is by far more efficient. However, increasing the number of factors may enhance the results, but is not economic.

We will focus on using more advanced methods for extracting keywords, in order to improve the matching process. Also, we aim to utilize some additional information related to venues, e.g., deadlines and locations. Bibliographic information also contains valuable data. For example, in some researches, citation information is considered a useful clue. We will also expect to launch this method as a web-based application to help researchers.

References

[1] F. Ricci, L. Rokach and B. Shapira (2011). Introduction to recommender systems handbook. Springer.

[2] D. Jannach, M. Zanker, A. Felfernig and G. Friedrich (2010). Recommender systems: An introduction. Cambridge University Press.

[3] J. Bobadilla, F. Ortega, A. Hernando and A. Gutiérrez (2013). Recommender systems survey. Knowledge-Based Systems 46:109–132.

[4] X. Su and T. M. Khoshgoftaar (2009). A survey of collaborative filtering techniques. Advances in Artificial Intelligence 2009:4.

[5] J. Beel, B. Gipp, S. Langer and C. Breitinger (2016). Paper recommender systems: A literature survey. International Journal on Digital Libraries 17(4):305–338.

[6] C. Pan and W. Li (2010). Research paper recommendation with topic analysis. In: Computer Design and Applications (ICCDA), 2010 International Conference on, IEEE, vol 4, pp V4–264.

[7] H. Luong, T. Huynh, S. Gauch, L. Do and K. Hoang (2012). Publication venue recommendation using author network's publication history. In: Asian Conference on Intelligent Information and Database Systems, Springer, pp 426–435.

[8] T. Huynh, K. Hoang and D. Lam (2013). Trend based vertex similarity for academic collaboration recommendation. In: International Conference on Computational Collective Intelligence, Springer, pp 11–20.

[9] C. Bancu, M. Dagadita, M. Dascalu, C. Dobre, S. Trausan-Matu and A. M. Florea (2012). ARSYS–article recommender system. In: Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2012 14th International Symposium on, IEEE, pp 349–355.

[10] M. Gori and A. Pucci (2006). Research paper recommender systems: A random-walk based approach. In: 2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2006 Main Conference Proceedings) (WI'06), IEEE, pp 778–781.

[11] Y. Jiang, A. Jia, Y. Feng and D. Zhao (2012). Recommending academic papers via users' reading purposes. In: Proceedings of the sixth ACM conference on Recommender systems, ACM, pp 241–244.

[12] J. Sun, J. Ma, X. Liu, Z. Liu, G. Wang, H. Jiang and T. Silva (2013). A novel approach for personalized article recommendation in online scientific communities. In: System Sciences (HICSS), 2013 46th Hawaii International Conference on, IEEE, pp 1543–1552.

[13] G. Cabanac (2011). Accuracy of inter-researcher similarity measures based on topical and social clues. Scientometrics 87(3):597–620.

[14] M. S. Pera and Y. K. Ng (2014). Exploiting the wisdom of social connections to make personalized recommendations on scholarly articles. Journal of Intelligent Information Systems 42(3):371–391.

[15] Z. Zhan, L. Yang, S. Bao, D. Han, Z. Su and Y. Yu (2011). Finding appropriate experts for collaboration. In: International Conference on Web-Age Information Management, Springer, pp 327–339.

[16] C. Basu, H. Hirsh, W. W. Cohen and C. Nevill-Manning (2001). Technical paper recommendation: A study in combining multiple information sources. Journal of Artificial Intelligence Research 14:241–262.

[17] D. Conry, Y. Koren and N. Ramakrishnan (2009). Recommender systems for the conference paper assignment problem. In: Proceedings of the third ACM conference on Recommender systems, ACM, pp 357–360.

[18] B. Gipp and J. Beel (2009). Citation proximity analysis (CPA)–a new approach for identifying related work based on co-citation analysis. In: Proceedings of the 12th International Conference on Scientometrics and Informetrics (ISSI'09), Rio de Janeiro (Brazil): International Society for Scientometrics and Informetrics, vol 2, pp 571–575.

[19] C. Caragea, A. Silvescu, P. Mitra and C. L. Giles (2013). Can't see the forest for the trees?: A citation recommendation system. In: Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries, ACM, pp 111–114.

[20] R. Klamma, P. M. Cuong and Y. Cao (2009). You never walk alone: Recommending academic events based on social network analysis. In: International Conference on Complex Sciences, Springer, pp 657–670.

[21] D. H. Park, H. K. Kim, I. Y. Choi and J. K. Kim (2012). A literature review and classification of recommender systems research. Expert Systems with Applications 39(11):10,059–10,072.

[22] G. H. Martín, S. Schockaert, C. Cornelis and H. Naessens (2013). An exploratory study on content-based filtering of call for papers. In: Information Retrieval Facility Conference, Springer, pp 58–69.

[23] E. Medvet, A. Bartoli and G. Piccinin (2014). Publication venue recommendation based on paper abstract. In: Tools with Artificial Intelligence (ICTAI), 2014 IEEE 26th International Conference on, IEEE, pp 1004–1010.

[24] Z. Xu, Y. Yang, F. Wang, J. Xu, Z. Li, F. Mu and L. Li (2014). A recommendation system for paper submission based on vertical search engine. In: Computer Engineering and Networking, Springer, pp 201–208.

[25] F. Beierle, J. Tan and K. Grunert (2016). Analyzing social relations for recommending academic conferences. In: Proceedings of the 8th ACM International Workshop on Hot Topics in Planet-scale Mobile Computing and Online Social Networking, ACM, pp 37–42.

[26] G. M. García, B. P. Nunes, G. R. Lopes, M. A. Casanova and L. A. P. P. Leme (2016). Comparing and recommending conferences. In: Proceedings of the 5th Bra, SNAM–Brazilian Workshop on Social Network Analysis and Mining.

[27] D. Peiris and R. Weerasinghe (2015). Citation network based framework for ranking academic publications and venues. In: Advances in ICT for Emerging Regions (ICTer), 2015 Fifteenth International Conference on, IEEE, pp 146–151.

[28] S. Perugini, M. A. Gonçalves and E. A. Fox (2004). Recommender systems research: A connection-centric survey. Journal of Intelligent Information Systems 23(2):107–143.

[29] X. Ren, Y. Zeng, Y. Qin, N. Zhong, Z. Huang, Y. Wang and C. Wang (2010). Social relation based search refinement: Let your friends help you! In: International Conference on Active Media Technology, Springer, pp 475–485.

[30] Y. Zeng, Y. Yao and N. Zhong (2009). DBLP-SSE: A DBLP search support engine. In: Web Intelligence and Intelligent Agent Technologies (WI-IAT'09), IEEE/WIC/ACM International Joint Conferences on, IET, vol 1, pp 626–630.

[31] D. Tchuente, M. F. Canut, N. Jessel, A. Peninou and F. Sèdes (2013). A community-based algorithm for deriving users' profiles from egocentrics networks: Experiment on Facebook and DBLP. Social Network Analysis and Mining 3(3):667–683.

[32] M. Eirinaki, J. Gao, I. Varlamis and K. Tserpes (2018). Recommender systems for large-scale social networks: A review of challenges and solutions.

[33] H. C. Wang and Y. L. Chang (2007). PKR: A personalized knowledge recommendation system for virtual research communities. Journal of Computer Information Systems 48(1):31–41.

[34] J. B. Lovins (1968). Development of a stemming algorithm. MIT Information Processing Group, Electronic Systems Laboratory Cambridge.

[35] C. D. Manning, P. Raghavan and H. Schtze (2008). Introduction to information retrieval. Cambridge University Press.

[36] T. K. Landauer and S. T. Dumais (1997). A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. Psychological Review 104(2):211.

[37] B. Sarwar, G. Karypis, J. Konstan and J. Riedl (2000). Application of dimensionality reduction in recommender system–a case study. Tech. rep., DTIC Document.

[38] M. Ley (2009). DBLP: some lessons learned. Proceedings of the VLDB Endowment 2(2):1493–1500.

Ramin Safa studied Computer Engineering at the Islamic Azad University of Lahijan, Guilan, Iran. In 2012 he got his B.Sc. degree and was accepted to the University of Guilan to earn a M.Sc. degree in Information Technology which he received in 2014. In 2015 he was accepted to pursue his studies as a full-time Ph.D. student in Software Systems at Islamic Azad University of Rasht. Today he is a Ph.D. candidate with research interests including Data Mining Techniques and Recommender Systems as well as their applications.

Seyed Abolghasem Mirroshandel received his B.Sc. degree from University of Tehran in 2005 and the M.Sc. and Ph.D. degree from Sharif University of Technology, Tehran, Iran in 2007 and 2012, respectively. Since 2012, he has been with Faculty of Engineering at University of Guilan in Rasht, Iran, where he is an Assistant Professor of Computer Engineering. Dr. Mirroshandel has published more than 50 technical papers in peer-reviewed journals and conference proceedings. His current research interests focus on Data Mining, Machine Learning, and Natural Language Processing.

Soroush Javadi received his B.Sc. degree in Software Engineering from University of Guilan, Rasht, Iran in 2015. His main research interests include Machine Learning and Natural Language Processing.

Mohammad Azizi received the B.Sc. and M.Sc. degree in Software Engineering from Islamic Azad University of Lahijan in 2012 and University of Guilan in 2016, respectively. His main research interests include Data Mining Techniques and Recommender Systems.

Eye Gaze Detection Based on Learning Automata by Using SURF Descriptor

Hasan Farsi*
Department of Electrical and Computer Engineering, University of Birjand, Birjand, Iran
hfarsi@birjand.ac.ir
Reza Nasiripour
Department of Electrical and Computer Engineering., University of Birjand, Birjand, Iran
reza.nasiripour@birjand.ac.ir
Sajjad Mohammadzadeh
Faculty of technical and engineering of Ferdows, University of Birjand, Birjand, Iran
s.mohamadzadeh@birjand.ac.ir

Received: 12/Nov/2017 Revised: 29/Apr/2018 Accepted: 19/May/2018

Abstract

In the last decade, eye gaze detection system has been known as one of the most important area activities in image processing and computer vision. The performance of eye gaze detection system is related to iris detection and recognition (IR). Iris recognition plays very important role for person identification. The aim of this paper is to achieve higher recognition rate compared to learning automata based methods. Usually, iris retrieval based systems consist of several parts including: pre-processing, iris detection, normalization, feature extraction and classification that are captured from eye region. In this paper, a new method without normalization step is proposed. Meanwhile, Speeded up Robust Features (SURF) descriptor is used to extract features of iris images. The descriptor of each iris image creates a vector with 64 dimensions. For classification step, learning automata classifier is applied. The proposed method is tested on three known iris databases; UBIRIS, MMU and UPOL database. The proposed method results in recognition rate of 100% for UBIRIS and UPOL databases and 99.86% for MMU iris database. Also, EER rate of the proposed method for UBIRIS, UPOL and MMU iris database are 0.00%, 0.00% and 0.008%, respectively. Experimental results show that the proposed learning automata classifier results in minimum classification error, and improves precision and computation time.

Keywords: Iris Retrieval; SURF; Learning Automata; Feature Extraction; Classification; Biometrics.

1. Introduction

A biometric system is based on unique features possessed by an individual. These features include: fingerprints, facial, voice, retina, iris and etc. Among these biometrics information, the system based on iris retrieval is more reliable and flexible for personal identification [1].

Eye gaze detection systems are based on iris retrieval. In generally, the iris retrieval system contains three steps. The first step is iris detection. The next step is locating the iris and the last step is a feature extraction from detected iris [2].

The performance of eye gaze detection systems depend on the extracted features from detected iris. There are many types of descriptor for feature extraction such as Histogram of Oriented Gradients (HOG), color, texture, Scale Invariant Feature Transform (SIFT), Principal Component Analysis (PCA) and etc. Boles et al applied zero-crossing representation of 1D wavelet transform for feature extraction [3]. The disadvantage of this method is to use 1D wavelet transform. This results in some drawbacks such as oscillations, shift variance, aliasing and lack of directionality [4]. In [5], Zhu et al presented 2D wavelet transform to extract features of iris images. However, 2D Wavelet transform is unable to resolve the aforementioned problems [6]. All natural signals are based on real-valued as speech, image and etc. Therefore, the reported method in [5] needed to use complex filtering. Montro et al reported a new method based on zero-crossing representation of 1D Discrete Cosine Transform [7]. They segmented iris image to its components by using Hough transform. The problem of this approach is to use 1D Discrete Cosine Transform (DCT). In the new methods, the 2D Discrete Cosine Transform along with zigzag scanning pattern are used which provide more information rather than 1D Discrete Cosine Transform [8]. Daugman applied 1D Gabor filter in feature extraction step [30]. In this method, the face is segmented as iris, gaze estimation and upper eyelid. The drawback of this method is as same as the reported method in [3]. In [9], the iris is separated by using circular Hough transform. In order to locate upper and lower eyelid, the authors applied Sobel edge detection operator. Belchar et al applied SIFT descriptor to extract features of iris [10]. The SIFT descriptor provides a feature vector with 128 dimensions. The high length of this vector corresponds to complexity.

The proposed system is evaluated on the three databases; UBIRIS [24] , UPOL [25] and MMU database

* Corresponding Author

[26]. In Figure 1, the examples of the iris images from these databases are shown.

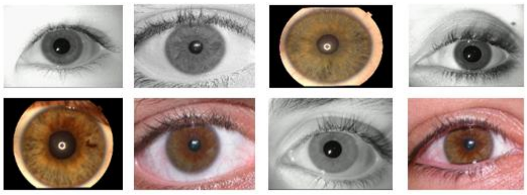


Fig. 1. Some example of UPOL, MMU and UBIRIS Database

This paper is organized as follows: in section 2, related works are discussed. Iris detection is explained in section 3. The proposed iris retrieval method is described in section 4. In this section, SURF descriptor, Learning Automata are explained. In section 5, experimental results are shown. Finally, the conclusion is drawn in section 6.

2. Related Works

Most of the reported method focuses on feature extraction and learns the features. In other words, after feature extraction, the next step is data learning. In this step, negative and positive images are used for training. Iris image is declared as positive and image without iris is considered as negative.

Liam et al [11] reported neural network for matching step. The authors used 150 samples for training. In this method, the authors extracted pupil by searching disk. The problem of this approach is to use disk, because size of pupil individual is different. Moinuddin et al [12] compared two different types of neural networks, MFNN and RBFNN. In [12], the edges are determined by using Sobel detector and then a feature vector is defined by iris boundary. The problem of this method is to use value of boundary as feature which is poor feature for training process in the algorithm. Ali and Salman [13] reported SVM classifier with different kernel types for iris retrieval. In this method, the features are extracted by Gabor wavelets. The disadvantage of this method is to use only the magnitude of Gabor filter output. Note that using both magnitude and phase of Gabor filter output provides higher recognition rate compared to using the magnitude alone [14]. In [15], Sarhan reported a method based on MLP neural networks. The method is followed by three-layer network. For feature extraction, this method exploits Discrete Cosine Transform (DCT). The problem of this method is as same as the reported approach in [5]. Fasca et al [16] exploited features by Local Binary Pattern (LBP) and HOG descriptors. The authors used Feed Forward Back Propagation Neural Network (FFBPNN) for training step. The authors applied two descriptors for feature extraction step which results in high computational complexity. Abiyer et al [17] applied neural network for iris retrieval system. The author described a gradient based learning model to learn their algorithm. In order to identify region of iris, they applied rectangular window with size of 10×10. The authors used same structure for iris detection. This results in high computational complexity by searching region of face. In [18], different types of learning algorithms for data classification are used such as Bayes, Euclidean, KNN probabilistic and non-probabilistic distance. Learning through kernel type corresponds to high computational complexity in learning step. In [19], iris retrieval method classified the iris images into multiple classes. The authors presented Principal Direction Divisive Partitioning (RDPP) to learn the iris images. For feature extraction, they proposed complex steerable pyramid. For iris detection, the authors suggested region of iris converted to 64 blocks. For each block, histogram is computed. Therefore, 24 histograms are defined as the features. The length of feature vector is too long and so results in high computational complexity. The reported method in [20] includes four stages. In first stage, the iris detection is applied on eye image to extract boundary between inner and outer contours. Then, iris image is segmented. In other words, the iris image is converted to new image with size of 16×16. The features are computed by using 2-D Gabor Wavelet Convolution. Finally, the reported approach is trained by Multi-dimensional artificial neural network (MDANN). The shortcoming of this method is as same as the reported method in [5]. In reported method in [21] is based on template matching. The segmentation is performed on captured iris image. Then for each iris image, the features are extracted by using Gabor filter. The authors generated iris template by encoding operation. Finally, the matching is performed between iris template and a new iris. The authors used Gabor filter which is unable to resolve same problem in [5]. Also, they used template matching for each iris image and therefore the test iris is compared to all irises image. This results in longer time in recognition step. The reported method in [22] performs the pre-processing operation on iris image. For feature extraction, texture feature is applied. In this step, the authors suggested using Local Binary Pattern (LBP) descriptor which provides five features for each iris as: Entropy, Variance, Inertia, the inverse of the contrast of the co-occurrence matrix (IDM) and Energy. They also used Gray Level Co-occurrence Matrix (GLCM) descriptor for iris image. Therefore, size of feature vector is 1×261. For iris retrieval, the algorithm is trained by probabilistic neural network. The problem of this method is to have long feature length. Therefore, this method needs much space to store the database of features. In reported method by Sachdeva and Kaur [44], iris is extracted from eye region by using iris template. Then, features are extracted from iris by Scale Invariant Feature Transform (SIFT). For learning step, SVM classifier is used. The precision of this method is 99.14% in IITD database. In [45], the authors used two classifiers, SVM and ANN classifier. For feature extraction, they applied 1D Log-Gabor wavelet technique. The precision obtained in UBIRIS database for ANN and SVM classifier are 92.5% and 95.9%, respectively.

The recent method is a conventional neural network (CNN). CNN is based on deep learning and consists of a number of hierarchical layers that provide unique features for each image. In these structures, CNNs are applied as iris segmentation [45]. Li and et al proposed a method that based on Convolutional Neural Networks (CNNs) [45]. For iris region, they used fully convolutional networks (FCN). The drawback of this method is applied fully image as input FCN networks. In other words, they ignored region of interest (ROI) for iris region. In [46], the different method compared to [45] was proposed for eye gaze detection. They used rough rule for ROI detection. In the next step and considering 21×21 mask, ROI was learned by using CNN.

Considering the mentioned problems, in this paper, a new method is proposed which provides a feature vector with length of 64 dimensions by SURF descriptor. Also, for training step, Learning Automata (LA) is used which is based on reinforcement learning. The main advantage LA compared to other methods is that the information is not required from the environment.

A block diagram of the proposed iris retrieval method is illustrated in Figure2. The first step is iris detection. In this part, the iris is identified from the eye or face image. The iris features are extracted by using SURF descriptor. The SURF descriptor is faster than SIFT descriptor and feature vector has 64 dimensions in length. LA classifier is used to train the proposed method. By using LA classifier, classification error in training phase tends to minimum. The obtained results from LA classifier shows that this method improves speeded up convergence, boundary separating between classes and reduces computation time [23].

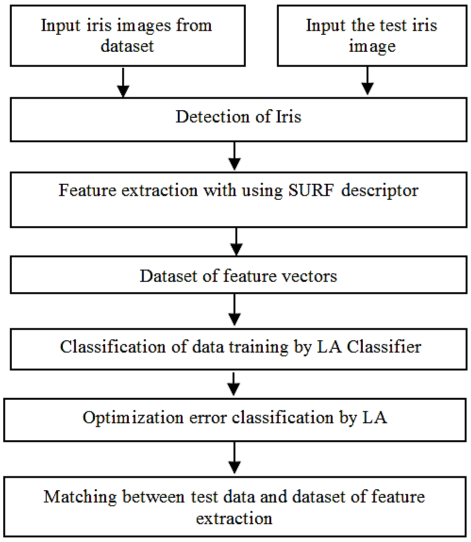

Input iris images from dataset
Input the test iris image
Detection of Iris
Feature extraction with using SURF descriptor
Dataset of feature vectors
Classification of data training by LA Classifier
Optimization error classification by LA
Matching between test data and dataset of feature extraction

Fig. 2. Block Diagram of the Iris Retrieval Method

3. Iris Detection Method

For iris detection, the image is pre-processed to remove extra information (such as noise and etc.). Next step is segmentation. In this step, the image is divided into multi-sections. In this paper, k-means algorithm is used to segment the image. Normally, the normalization step is considered after segmentation step. However, in this paper, the normalization step is removed because we obtain same results with and without normalization. By removing this step, the computation time of the proposed method is improved. Next step is edge extraction of the iris image. We have designed for multi-scale and multi-directions [27]. Gabor wavelet is used to detect the edges. By changing the parameters of this wavelet, different scales are achieved. For creating each scale of iris image, Gabor filter is convolved with the original iris image in -90 to +90 degrees (Equations (1) [27]):

$$R(x-x_c, y-y_c)_{\lambda,\sigma,\theta,\varphi,\gamma} = R_0 \exp\left(-\frac{u^2+,\gamma^2 v^2}{2\sigma^2}\right) \cos\left(2\pi\frac{u}{\lambda}\varphi\right) \quad (1)$$

$$u(x-x_c, y-y_c, \theta) = (x-x_c)\cos\theta - (yy_c)\sin\theta$$

$$v(x-x_c, y-y_c, \theta) = (x-x_c)\sin\theta + (yy_c)\cos\theta$$

In this equation, x_c and y_c are the rotation centers of the filter to the preferred angle, θ, that are placed related to the origin. σ, λ and φ are standard deviation, length wave and filter phase difference, respectively. After creating the edges by using different scales, these scales are combined together to present comprehensive edge map. Figure 3 shows the illustration of the iris detection method.

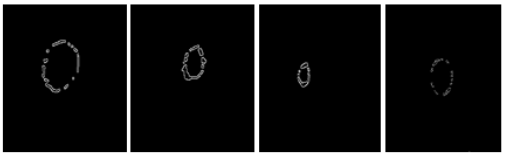


Fig. 3. The Results of Iris Detection Methods

4. Proposed Iris Retrieval Method

In this section, the iris retrieval method is explained. After iris detection, the features are extracted from the iris image. In this paper, SURF descriptor [28] is used which is later introduced in section 4.1. Then, the database of the feature vectors is created. According to this database, matching process is performed between the test image and all images. Next step is training. In this step, the LA classifier is used. By using Learning Automata, the classification error is optimized. In section 4.2, classification and optimization based on Learning Automata are introduced.

4.1 SURF Descriptor

The SURF algorithm reported in [28] is applied for four steps:

a. Scale-space extreme detection
b. Key point localization
c. Orientation assignment
d. Key point descriptor

In the first step, this descriptor uses the determinant Hessian matrix to find candidate points. The points considered as candidate points which have no changes against occlusion and orientation. The hessian matrix, $H(x,\sigma)$, at scale, σ, in x is defined as [28]:

$$H(x,\sigma) = \begin{bmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{bmatrix} \tag{2}$$

Where $L_{xx}(x,\sigma)$, $L_{xy}(x,\sigma)$ and $L_{yy}(x,\sigma)$ denote the convolution of the image at point of $X(x,y)$. By changing the parameter, σ, different scales are achieved. In next step, each candidate point is compared to 8 points in the same scale, 9 points in the upper scale and 9 points in the lower scale. The point is considered as key point in which has extreme value between 26 neighbors of scales. After localization of the key points, orientation assignment is considered for each key point. The dominant orientation is estimated by calculating sum of the horizontal and vertical Haar wavelet responses within a sliding orientation window with angle of $\pi/3$.

Final step is constructed by a square window along with the dominant orientation with size of 20σ. This window is subsequently divided into 4×4 regular sub-regions. Then, for each sub-region, Haar wavelet responses are calculated. As shown in Figure 4, for the window containing 4×4 sub-region, each feature point can be described using a 64-dimensional vector. Figure 5 shows illustration of the iris image with extracted features by using SURF descriptor.

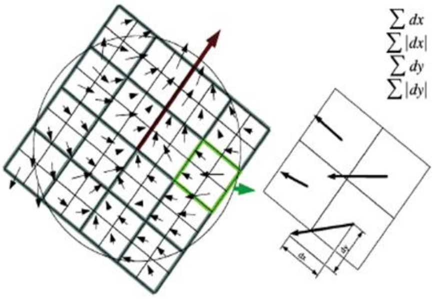


Fig. 4. Computed Features Vector

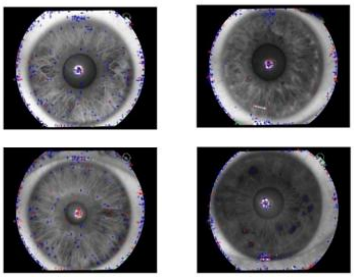


Fig. 5. Feature Extraction by SURF Descriptor

4.2 Learning Automata

4.2.1 Concept of Learning Automata

For the first time, learning automata (LA) was reported by Testlin in 1960s [29]. The LA can be considered as a single object which has a finite number of actions. Generally, the LA works by choosing an action from a set of actions and then this action is applied on the environment. This selection is evaluated by the random environment and for the next selection, the automata is used based on environment response. During this process, the automaton learns to choose its optimal action. During the last two decades, the LA has been widely used by researchers. For example in the pattern recognition area, it is suggested to use neural network for automation operation [30]. The authors reported NN[1] with arrange 9-3-1. In this arrange, 9 refers to the number of neurons input layer, 3 represents neurons middle layer and 1 indicates neuron of output layer.

In fact, learning automata is reinforcement learning. The main advantage of the LA compared to other methods is that no information is required from the environment. In supervision based methods, inputs and targets are already determined, but in reinforcement learning, automata must learn oneself. Reinforcement learning allows the method agent to learn its behavior based on feedback from the environment. This behavior can be learnt once for all, or kept on adapting by passing the time. If the problem is accurately modelled, some reinforcement learning algorithms can converge to the global optimum; this is the ideal behavior that maximizes the reward. This automated learning scheme implies that there is little need for a human expert who knows about the domain of application. Much less time will be spent for designing a solution, since there is no need for hand-crafting complex set of rules as expert system, and all that is required is someone who is familiar with Reinforcement learning.

In general, LA takes input β and changes its mode by using an internal function [31]. Then, output α is delivered to environment. Each learning automaton is characterized by a set of internal states, input actions or set of inputs, state probability distributions, and reinforcement scheme or set of outputs, which are connected in a feedback loop to the environment, as shown in Fig. 6 [30].

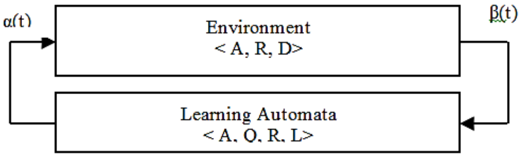


Fig. 6. Learning Automata in the Environment [30]

The learning is defined as a change in behavior resulted from past experience with the passage of time. At each step, an action is selected based on probability distribution. The environment responses to the selected action and accordingly sends a response to LA with either a reward or a penalty. By repeating these interactions, the automaton converges to the optimal action. In LA based methods, parameters are defined as follows:

- $A = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ Presents set of actions with total number of r actions.
- R: This is limit for responses due to the environment.
- $D = \{d_1, d_2, \dots, d_r\}$ Presents a set of reward or penalty probability.

[1] Neural Network

- Q: Presents state of automation by $(k) = \{p(k), \hat{D}(k)\}$. Where $p(k)$ is the action probability and $\hat{D}(k)$ is vector of reward or penalty probability.
- L: Presents the learning algorithm.

In this algorithm, $p(k)$ is changed in each iteration. If response of environment is penalty then:

$$p_j(k+1) = \begin{cases} p_j(k) + a(1 - p_j(k)) & \forall j = i \\ (1-a)p_j(k) & \forall j \neq i \end{cases} \quad (3)$$

Where a is reward parameter. If response of environment is reward, $p(k)$ is updated by Eq. 4:

$$p_j(k+1) = \begin{cases} (1-b)p_j(k) & \forall j = i \\ \left(\frac{b}{r-1}\right) + (1-b)p_j(k) & \forall j \neq i \end{cases} \quad (4)$$

Where b is penalty parameter.

Based on the relation between a, b, learning methods are defined as follows:

- If $a = b$, learning method is defined as Linear Reward Penalty (L_{R-P}).
- If $a \gg b$, learning method is defined as Linear Reward Epsilon Penalty ($L_{R-\varepsilon P}$).
- If $b = 0$, learning method is defined as Linear Reward Inaction (L_{R-I}).

4.2.2 Clustering Based On Learning Automata

In this section, clustering process is performed by using LA. After feature extraction, the feature vectors are stored in a database as features database. Then, a number of these vectors are considered as data training. Based on these vectors, training step is started. In this paper, we apply LA as shown in Figure 7. For each iris image and the length of SURF descriptor feature vector has 64 dimensions.

In this paper, LA classifier reported by S.H. Zahiri is used [23]. Based on feature vector, decision hyper plane is considered as follows:

$$d(x) = w_1x_1 + w_2x_2 + \cdots + w_{64}x_{64} + w_{65} \quad (5)$$

Where $X = (x_1, x_2, \dots, x_{64}, 1)$ denotes the features which are extracted by SURF descriptor and $W = (w_1, w_2, \dots, w_{64}, w_{65})$ is called weighted vector.

Normally, log_2^M is the number of decision hyper planes (M denotes the number of classes) which separates classes from each other. A data belongs to ith class if:

$$d_i(x) = w'_iX > 0 \qquad i = 1,2, \dots, M \quad (6)$$

Where $d_i(x)$ denotes ith decision hyper plane and w_i is the weighted vector for ith decision hyper plane.

So far, the classes have been separated from each other. In this paper, the numbers of decision hyper planes based on separated classes are considered according to Eq. 7:

$$X\varepsilon C_i if \quad d_i(X) > d_j(X) \quad for\ all \quad j \neq i \quad (7)$$

Where C_i is ith class.

LA classifier provides the best vector weights for decision hyper planes. According to weight vector variable, $W_i(i = 1,2, \dots, H)$, where H is the number of decision hyper plane, fitness function for data of iris is defined as:

$$f(W) = R(1 - \beta M) \quad (8)$$

Where R is the number of correct classified, β is a parameter which is experimentally obtained and M denotes the number of misclassified. In this paper, we consider $\beta = 0.5$.

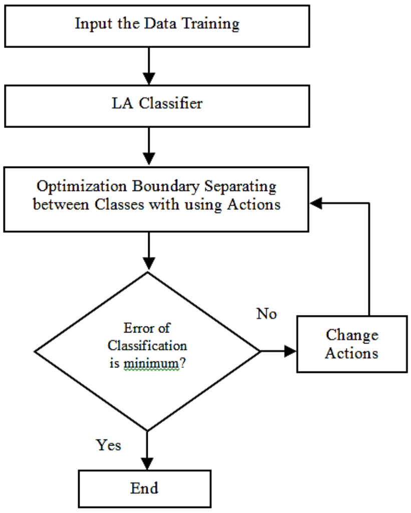

Input the Data Training
LA Classifier
Optimization Boundary Separating between Classes with using Actions
Error of Classification is minimum?
No
Change Actions
Yes
End

Fig. 7. Block Diagram Classification and Optimization by LA

The aim of the LA classifier is optimization of separating boundary between the classes or minimization of classification error, as shown in Fig. 7. In the following, the structure of the LA classifier is explained.

4.2.3 The Structure of LA Classifier

According to above descriptions, the LA classifier is based on minimization of Equation 6 which is performed in following steps [23]:

Step 1: Initialization of Internal Parameters

γ: Number of hyper planes
δ: Threshold of action probabilities
s: Normalized factor of convergence
ε: Error band

Step 2: Display of r Hyperplanes on Feature Space

According to each action, parameters are defined as:

- $\eta_i(n)$: Total rewards or penalties obtained by the action α_i.
- $z_i(n)$: Number of times in which the action α_i is chosen
- $\xi_i(n) = \frac{\eta_i(n)}{z_i(n)}$
- $\xi_m(n) = Max_i\{\xi_i(n)\}$
- $\xi_l(n) = Min_i\{\xi_i(n)\}$

$p(n)$: Action probability distribution of α_i

Step 3: Search Loop

- Repeat
- Pick up an action $\alpha(n) = \alpha_i(n)$ according to $p(n)$

- Randomly select a set of decision weight vector
- Calculate Equation 8
- Update $\xi_i(n)$ as follows:
- If $\alpha(n) = \alpha_i$, Then

$$\eta_i(n+1) = \eta_i(n) + \frac{T - f(W)}{T}$$

Where T is the total of training data.

- $z_i(n+1) = z_i(n) + 1$
- $\xi_i(n+1) = \frac{\eta_i(n+1)}{z_i(n+1)}$

For all $j \neq i$

- $\eta_j(n+1) = \eta_j(n)$
- $z_j(n+1) = z_j(n)$
- $\xi_j(n+1) = \xi_j(n)$

Update $p(n)$ as follows:

- $p(n+1) = \left(1 - s \times \xi_m(n)\right) \times p(n) + s \times \xi_m(n)$
- If $P_l(n) = Min_i\{p_i(n)\} < \delta$, Then Go to the next step
- Else, $n = n + 1$
- End Repeat.

This process continues until the classification error is minimized. Then, the best actions, weighted vectors, are stored as decision hyper planes. Figure 8 shows illustration of LA classifier and optimization on MMU, UBIRIS and UPOL database. As observed in Figure 8, for three databases, the convergence is occurred in few numbers of iterations. The error of classification is related to UBIRIS database, but the convergence of this database in compared to MMU and UPOL database has occurred later.

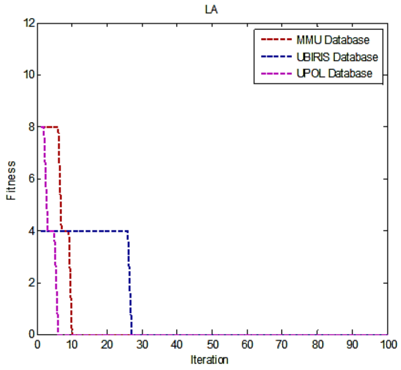


Fig. 8. The result of the LA classifier and optimization for three databases; MMU, UBIRIS and UPOL iris database

5. Results

The proposed method based on learning automata is implemented by MATLAB (version 8.1) with configuration as follows: processor: Intel core i5, OS: Windows 8, CPU speed: 2.50 GHz and RAM: 6 GB.

In this section, the performance of the proposed method is evaluated on three databases, UBIRIS, MMU and UPOL. The details of these databases are listed in Table 1. The MMU database is composed of 45 topics in which each topic contains 10 eye images, 5 eye images for left of eye and 5 images for right of eye [26]. The UBIRIS database is composed of 241 subjects [24]. The total numbers of image are 1877 images. The original size for each iris image is 800 × 600. The UPOL iris database is composed of 64 topics. Each topic includes six images, three left iris images and three right iris images [25]. Size of iris image is 768 × 576.

Table 1. The Number of iris images, size and the number of subjects

Database	Original size iris image	Number of iris image	Number of subject
MMU	320×280	450	45
UBIRIS	800×600	1877	241
UPOL	768×480	384	64

Meanwhile, for evaluation of the proposed iris retrieval method, we use two measures, Equal Error Rate (EER) and recognition rate. The EER is biometric measure that is composed of False Rejection Rate (FRR) and False Acceptance Rate (FAR) [32].

$$FRR = \frac{Number\ of\ misclass\ ified\ as\ iris\ samle}{Total\ Number\ of\ data\ in\ database} \tag{9}$$

$$FAR = \frac{Number\ of\ iris\ samle\ as\ misclass\ ified}{Total\ Number\ of\ data\ in\ database} \tag{10}$$

When FRR and FAR are equal, the obtained value is defined as EER. The reliable performance of iris retrieval occurs when EER is very low. In following, the experimental results for MMU, UBIRIS and UPOL databases are explained.

5.1 The MMU Database

We have compared the proposed iris retrieval to the reported algorithms in [33-34-35-36-22-48]. In [33], Elgamal et al reported a method based on DWT and PCA[1]. The authors used 2/3 images for training part and rest images for testing part. Using 1D DWT caused to extract poor feature compared to 2D DWT. The shortcomings of DWT are oscillations, shift variance, aliasing and lack of directionality [4]. In [34] and [16], Kumar et al used Haar wavelet and logarithm Gabor filter. They reported new sets for training with a number of variables. The disadvantage of this method is that the authors used only magnitude part of Gabor wavelet output. However, using magnitude and phase of Gabor filter output, provides higher recognition rate [44]. In [35], Rahulkar et al reported a new method based on triplet half-band filter bank. In this method, two samples are considered for training and three samples for testing. In [36], Baqar et al provided dual boundary contour vector. Using iris boundaries as features results in poor feature for training step. This method used three images for training and two images for testing. In [48], the author used different

[1] Principal Component Analysis

descriptor, Gabor, Riesz and Taylor. They also applied mixed descriptor to create new descriptor.

The recognition rate and EER measures resulted by the proposed method and the reported methods in [33-34-35-36-22] on the MMU database are presented in Table 2. As observed, the value of EER rate by the proposed method, Elgamal et al, Kumar et al, Rahulkar et al, Barqar et al and Hajari et al are 0.008%, 0.040%, 2.590%, 1.880%, 0.023% and 1.530%, respectively. Also, the recognition rate for the proposed method is 99.86%. Therefore, the proposed method provides the best performance in MMU database.

Table 2. Comparison of Recognition and EER rate in MMU database

Method	Recognition rate (%)	EER rate (%)
Elgamal et al [33]	99.50	0.040
Kumar et al [34]	81.37	2.590
Rahulkr et al [35]	87.18	1.880
Barqar et al [36]	99.00	0.023
Hajari et al [22]	95.50	1.53
Gabor [48]	85.50	--
Taylor [48]	97.50	--
Gabor+Taylor [48]	97.66	--
The proposed method	99.86	0.008

5.2 The UBIRIS Database

In this paper, we have used first session of this database due to having good quality images. In this sub-section, we compare the proposed method to the reported methods in [37-38-39-40]. The reported methods in [37], [38] and [39] used GLCM[1] based Haralic features, Gabor features with kernel Fisher and Gabor filter, respectively. The GLCM descriptor is based on texture analysis for iris image. The reported method in [40] is based on Gabor filter and uses only the magnitude. In [37] and [39], three samples are used in training step and two samples for testing part. In this database, we also use three samples for training and two samples for testing.

As observed in Table 3, the value of recognition rate by the proposed method, Sundaram et al, Tallapragada et al, Tsai et al, Naresh and reported method in [48] are 100%, 97.00%, 96.60%, 97.20%, 79.90%, 80%, 85%, 85%, 92.50% and 95.90%, respectively. As observed, the values of EER and recognition rate by the proposed method are 0.00% and 100%, respectively. The obtained results show that the performance of the proposed method is better than the other methods.

Table 3. Comparison of Recognition and EER rate in UBIRIS database

Method	Recognition rate (%)	EER rate (%)
Sundaram et al [37]	97.00	7.09
Tallapragada et al [38]	96.60	8.19
Tsai et al [39]	97.20	7.80
Narseh et al [40]	79.90	8.93
Gabor [48]	80.00	--
Taylor [48]	85.00	--
Gabor+Taylor [48]	85.00	--
ANN [45]	92.50	
SVM [45]	95.90	
The proposed method	100	0.00

5.3 The UPOL Database

In this database, we have compared the proposed method against the reported methods in [19-41-42-47]. In [19], Ross et al proposed complex steerable pyramid. The authors used the number variables of iris image for training. The authors in [41] and [42] proposed Coiflet wavelet transform and Haar, Symlet, biorthogonal, respectively. They presented three iris images for training and two iris images for testing part. In [47], the author used different descriptors to learn features. Also, theses descriptors are combined together to create new descriptor. In this database, we also use three samples for training and two samples for testing. As observed in Table 4, the recognition rate by the proposed method is 100%. As shown, the value of EER rate by the proposed method, Ross et al, Harjoko et al and Masood are 0.00%, 0.00%, 0.28% and 0.04%, respectively.

Table 4. Comparison of Recognition and EER rate in UPOL database

Method	Recognition rate (%)	EER rate (%)
Ross et al [19]	100	0.00
Harjoko et al [41]	82.90	0.28
Masoud et al [42]	95.90	0.04
HOG + KNN [47]	99.12	0.016
HOG + SVM [47]	87.14	0.18
LBP + KNN [47]	97.14	0.072
LBP + SVM [47]	90.69	0.12
The proposed method	100	0.00

The proposed method is also compared to the reported method in [43]. In [43], the authors reported a method in which multi-scale morphologic operator is used for feature extraction. In this experiment, iris retrieval is applied on two databases, UPOL and UBIRIS iris databases, for left and right iris images, and the obtained results are presented in Table 5. For example, the value of recognition rate left iris on UBIRIS database by the proposed method and Umer et al are 98.18%, 85.40%, respectively.

Table 5. Comparison of recognition rate left and right iris

Database	Method	Recognition rate (%)	
		L	R
UBIRIS	Umer et al [43]	85.40	91.56
	The proposed method	98.18	95.23
UPOL	Umer et al [43]	89.06	84.38
	The proposed method	97.51	85.19

One of the important measures in eye gaze detection is cost computational. We simulated by MATLAB (version 8.1) with configuration as follows: processor: Intel core i5, OS: Windows 8, CPU speed: 2.50 GHz and RAM: 6 GB. The cost computational on the MMU, UBIRIS and UPOL database are presented in Table 6.

Table 6. Comparison of the computational cost by the proposed method in terms of millisecond (ms).

The computational cost	
UPOL	52.10
MMU	56.11
UBIRIS	49.74

[1] Gray Level Co-occurrence Matrix

6. Conclusions

In this paper, a new method based on learning automata was proposed. The SURF descriptor was used for feature extraction. By using SURF descriptor, the proposed method improves performance, computation time and reduces the required storage space. Also, the LA classifier was applied for separating decision boundary. Meanwhile, by using LA, the classification error was minimized and tends to zero. The proposed method was compared to other methods on three databases including UBIRIS, UPOL and MMU iris databases. The obtained results show that the performance of the proposed method is better than other methods.

References

[1] Z. Zhu, Q. Ji, "Robust real-time eye detection and tracking under variable lighting conditions and various face orientations", In Computer Vision and Image Understanding, Vol. 98, Vo. 1, 2005, pp. 124-154.

[2] L. Ma, Y. Wang, T. Tan., "Iris recognition based on multichannel Gabor filtering", Proc. Fifth Asian Conf. Computer Vision, 2002, pp. 279-283.

[3] W. Boles, W., B. Boashash, "A human identification technique using images of the iris and wavelet transform", IEEE transactions on signal processing, Vol. 46, No. 4, 1998, pp. 1185-1188.

[4] A. Sović, D. Seršić., "Robustly adaptive wavelet filter bank using L1 norm", in Systems, Signals and Image Processing (IWSSIP), 18th International Conference on. IEEE, 2011.

[5] Y. Zhu, T. Tan, Y. Wang, "Biometric personal identification based on iris patterns. in Pattern Recognition", Proceedings. 15th International Conference on. IEEE, 2000.

[6] T. Bulow, G. Sommer, "Hypercomplex signals-a novel extension of the analytic signal to the multidimensional case", IEEE Transactions on signal processing, Vol. 49, No. 11, 2001, pp. 2844-2852.

[7] D. Monro, S. Rakshit, D. Zhang, "DCT-based iris recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 29, , No. 4, 2007.

[8] T. Pradeepthi, A.P. Ramesh. "Pipelined architecture of 2D-DCT, Quantization and Zigzag process for JPEG image compression using VHDL", International Journal of VLSI Design & Communication Systems, Vol. 2, No. 3, 2011.

[9] R. Ng, Y.H. Tay, K.M. Mok, "An effective segmentation method for iris recognition system", IET image processing, 2008.

[10] C. Belcher, Y. Du, "Region-based SIFT approach to iris recognition", Optics and Lasers in Engineering, Vol. 47, 2009, No. 1, 2009, pp. 139-147.

[11] L. Liam, A. Chekima, LC. Fan, J.A, Dargham, "Iris recognition using self-organizing neural network", in Research and Development, 2002.

[12] M. Moinuddin, M. Deriche, S.S.A. Ali, "A New Iris Recognition Method based on Neural Networks", WSEAS Transactions on information science and applications, 2004.

[13] H. Ali, M.J. Salami, "Iris recognition system using support vector machines", in Biometric Systems, Design and Applications. InTech, 2004.

[14] W. Zhang, S. Shan, L. Qing, X. Chan, W. Gao, "Are Gabor phases really useless for face recognition?", Pattern Analysis and Applications, Vol. 12, No. 3, 2009, pp. 301-307.

[15] A.M Sarhan, "Iris Recognition Using Discrete Cosine Transform", Journal of Computer Science, Vol. 5,No. 5, 2009, pp. 369-373.

[16] P.F.G Mary, P.S.K. Paul, J. Dheeba, "Human identification using periocular biometrics", International Journal of Science, Engineering and Technology Research (IJSETR) , 2013.

[17] R.H Abiyev, K. Altunkaya., "Personal iris recognition using neural network", International Journal of Security and its Applications, Vol. 2, No. 2, 2008, pp. 41-50

[18] P.C Murty, E.S. Reddy, "Iris recognition system using principal components of texture characteristics", TECHNIA-Int. J. Computing Science and Communication Technologies, Vol. 2, No. 1, 2009, pp. 343-348.

[19] A. Ross, M.S. Sunder, "Block based texture analysis for iris classification and matching", in Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE Computer Society Conference on, 2010.

[20] R. Farouk, R. Kumar, K. Riad., "Iris matching using multi-dimensional artificial neural network", IET Computer Vision, Vol. 5,No. 3, 2011, pp. 178-184.

[21] A. Varshney, A. Rani, V. Singh., "Optimization of filter parameters for iris detection", in Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), 2015.

[22] K. Hajari, U. Gawande, Y. Golhar., "Neural Network Approach to Iris Recognition in Noisy Environment", Procedia Computer Science, 2016, pp. 675-682.

[23] S.H Zahiri, "Learning automata based classifier", Pattern Recognition Letters, Vol. 29, 2008, No. 1, 2008, pp. 40-48.

[24] http://iris.di.ubi.pt/ubiris1.html

[25] http://www.cbsr.ia.ac.cn:8080/iapr_database.jsp

[26] http://pesona.mmu.edu.my/

[27] H. Farsi, R. Nasiripour, S. Mohamadzadeh., "Improved Generic Object Retrieval In Large Scale Database By SURF Descriptor", Journal of Information Systems and Telecommunication (JIST). Vol. 5, No. 2, 2017, pp. 128-137.

[28] H. Bay, A. Ess, T. Tuytelaars, L.V. Gool, "Speeded-up robust features (SURF)", Computer vision and image understanding, Vol. 110, No. 3, 2008, pp. 346-359.

[29] M. obaidat, G. Papadimitiou, A. Pomportsis, "Guest Editorial Learning Automata: Theory, Paradigms and Applications", IEEE Transactions on systems, man, and cybernetics, Vol. 32. No. 6, 2002.

[30] Thathachar, M.A. and P.S. Sastry., "Varieties of learning automata: an overview", IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), Vol. 32, No. 6, pp. 711-722, 2002.

[31] K.S Narendra, M.A. Thathachar., "Learning automata: an introduction", Courier Corporation, 2012.

[32] D.D Zhang, "Automated biometrics", Technologies and systems. Vol. 7.: Springer Science & Business Media, 2007.

[33] M. Elgamal, N. Al-Biqami, "An efficient feature extraction method for iris recognition based on wavelet transformation", Int. J. Comput. Inf. Technol, Vol. 2, No. 03, 2013, pp. 521-527.

[34] A. Kumar, A. Passi, "Comparison and combination of iris matchers for reliable personal authentication", Pattern recognition. Vol. 43, No. 3, 2013, pp. 1016-1026.

[35] A.D. Rahulkar, R.S. Holambe, "Half-iris feature extraction and recognition using a new class of biorthogonal triplet half-band filter bank and flexible k-out-of-n: a postclassifier", IEEE Transactions on Information Forensics and Security, Vol. 7, No.1, 2012, pp. 230-240.

[36] M. Baqar, A. Azhar, Z. Lqbal and et al., "Efficient iris recognition system based on dual boundary detection using robust variable learning rate Multilayer Feed Forward neural network", in Information Assurance and Security (IAS), 7th International Conference on. 2011.

[37] R.M Sundaram, B.C. Dhara, "Neural network based Iris recognition system using Haralick features. in Electronics Computer Technology (ICECT), 2011.

[38] V. Tallapragada, E. Rajan, "Improved kernel-based IRIS recognition system in the framework of support vector machine and hidden Markov model", IET image processing, Vol. 6, No. 6, 2012, pp. 661-667.

[39] C.C Tsai, et al., "Iris recognition using possibilistic fuzzy matching on local features", IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), Vol. 42, No. 1, , 2012, pp. 150-162.

[40] N.N Babu, V. Vaidehi, "Fuzzy based IRIS recognition system (FIRS) for person identification", in Recent Trends in Information Technology (ICRTIT), International Conference on. 2011.

[41] A. Harjoko, S. Hartati, H. Dwiyasa, "A method for iris recognition based on 1d coiflet Wavelet", world academy of science, engineering and technology, Vol. 56, No. 24, 2009, pp. 126-129.

[42] K. Masood, M.Y. Javed, A. Basit., "Iris recognition using wavelet", in Emerging Technologies, ICET. International Conference on. 2007.

[43] S. Umer, B.C. Dhara, B. Chanda., "Iris recognition using multiscale morphologic features", Pattern Recognition Letters, 2015, pp. 67-74.

[44] G. Sachdeva, B. Kaur, "Iris Recognition Using Fuzzy SVM Based On SIFT Feature Extraction Method", in International Journal of Modern Computer Sience (IJMCS), Vol. 4, No. 2, 2016, pp. 16-22.

[45] S. Salve, S. Narote, "Iris recognition using SVM and ANN", in Wireless Communications, Signal Processing and Networking (WiSPNET), International Conference on. 2016. IEEE.

[46] N. Liu, H. Li, M. Zhang, J. Liu, Z. Sun, T. Tan, "Accurate iris segmentation in non-cooperative environments using fully convolutional networks", In Proceedings of the IEEE International Conference on Biometrics, Halmstad, Sweden, 2016, pp. 1–8.

[47] M. Arsalan, H. Gil, R. Naqvi, M. Lee, M. Kim, D. Kim, C. Sik, K. Park, "Deep Learning-Based Ireis Segmentation for iris recognition in visible light environment", in MDPI Journal, 2017.

[48] M. Alhamrouni, "Iris Recognition By Using Image Processing Techniques", A thesis submitted to the Graduate School of Natural And Applied Science of ATLIM University, 2017.

[49] B. Shekar, S. Sharada, "Multi-patches iris based person authentication system using particle swarm optimization and fuzzy c-means clustering", in The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2017.

Reza Nasiripour was born in Mashhad in 1990. He received the B.Sc. and M.Sc. degrees in electrical communication engineering from University of Birjand, Birjand, Iran in 2012 and 2014, respectively. He is currently Ph.D. student in Department of Electrical and Computer Engineering, University of Birjand, Birjand, Iran. His research interests include Image and Video Processing, Pattern Recognition, Machine Learning and Deep Learning.

Hassan Farsi received the B.Sc. and M.Sc. degrees from Sharif University of Technology, Tehran, Iran, in 1992 and 1995, respectively. Since 2000, he started his Ph.D in the Centre of Communications Systems Research (CCSR), University of Surrey, Guildford, UK, and received the Ph.D degree in 2004. He is interested in speech, image and video processing on wireless communications. Now, he works as professor in communication engineering in department of Electrical and Computer Eng., University of Birjand, Birjand, IRAN.

Sajad Mohamadzadeh received the B.Sc. degree in communication engineering from Sistan & Baloochestan, University of Zahedan, Iran, in 2010. He received the M.Sc. and Ph.D. degree in communication engineering from South of Khorasan, University of Birjand, Birjand, Iran, in 2012 and 2016, respectively. Now, he works as assistant professor in Faculty of Technical and Engineering of Ferdows, University of Birjand, Birjand, Iran. His area research interests include Image and Video Processing, Retrieval, Pattern recognition, Digital Signal Processing, Sparse Representation, and Deep Learning.

Effective Solving the One-Two Gap Problem in the PageRank Algorithm

Javad Paksima
Department of Engineering, Payame Noor University, Tehran , Iran
Paksima@pnu.ac.ir
Homa khajeh*
Department of Engineering, Science and art University, Yazd, Iran
khajeh121@yahoo.com

Received: 06/Sep/2017 Revised: 17/Dec/2017 Accepted: 08/Apr/2018

Abstract

One of the criteria for search engines to determine the popularity of pages is an analysis of links in the web graph, and various methods have already been presented in this regard. The PageRank algorithm is the oldest web page ranking methods based on web graph and is still used as one of the important factors of web pages on Google. Since the invention of this method, several bugs have been published and solutions have been proposed to correct them. The most important problem that is most noticed is pages without an out link or so-called suspended pages. In web graph analysis, we noticed another problem that occurs on some pages at the out degree of one, and the problem is that under conditions, the linked page score is more than the home page. This problem can generate unrealistic scores for pages, and the link chain can invalidate the web graph. In this paper, this problem has been investigated under the title "One-Two Gap", and a solution has been proposed to it. Experimental results show that fixing of the One-Two gap problem using the proposed solution. Test standard benchmark dataset, TREC2003, is applied to evaluate the proposed method. The experimental results show that our proposed method outperforms PageRank method theoretically and experimentally in the term of precision, accuracy, and sensitivity with such criteria as PD, P@n, NDCG@n, MAP, and Recall.

Keywords: One-Two Gap; PageRank; Search Engine; Web Graph.

1. Introduction

Search has become the predominant way of getting our everyday information in Web. Since online resources are growing rapidly, the use of search tools is required. 91% of search engine users said that when they use the search engine, they usually or more often find the information they need [1]. According to a study conducted in [2], users only examine the results of the top rankings, indicating the importance of ranking. Unit ranking is one of the most important parts of the search engine. Ranking is a process by which the page quality is estimated by the search engine. Currently, there are two major methods to rank web pages. In the first method, the ranking is based on the content of the web page (traditional ranking). Models such as the Probabilistic, Vector Space, and Boolean models are presented for content-based ranking [3]. The second method determines the importance of ranking pages based on web graph and web connections.

Unlike the traditional information retrieval environment, the Web has a large heterogeneous structure, with web pages attached to each other and forming a large graph. Web links include valuable information [4]; therefore, new ranking algorithms are created based on the link. Their main strength is to use the contents of other pages to rank a page [5].

Most search engines use algorithms to score pages based on the web graph. Links represent the quality of a page's content from the perspective of the outer pages (as opposed to the textual content of the page that is fully dependent on its creator). The link text usually contains a descriptive description of a page by other pages; in other words, the ranking is based on a link from the content of other pages to evaluate a page. Most graph-based methods are designed with the assumption that links are created by someone other than the page designer, and the purpose is to advise the page, but this is not always the case.

These algorithms are divided into two major categories: independent of queries, dependent on queries. In query-dependent methods such as PageRank [5] and HostRank [6], ranking is online and using the entire web graph. As a result, the rank of each page is constant for each query; but in query-based methods like HITS [7], ranking is performed only in part of the web graph, which includes query-related pages.

Among the algorithms, the PageRank algorithm is more important because the only algorithm in the search engine is to rank web pages [8]. It is currently used by Google's renowned search engine. Almost every algorithm is presented in a ranking, which has a problem, and PageRank is no exception to this rule. Some of the PageRank problems are addressed in Section 1-2. We encountered a new problem in examining the Web graph and calculating PageRank; that is, if a page has only one backlink, the second page may have a higher score than the first one, which applies to pages with double-top

* Corresponding Author

output levels That is why the name of the problem was "One-Two Gap".

The rest of the article is organized as follows: The PageRank algorithm, and its problems are discussed and used terms in this article are expressed in Section 2. In Section 3, the One-Two Gap problem will be explained. In Section 4, we will analyze this problem analytically and identify the pages that have this problem. Section 5 provides a solution to this problem; and in Section 6, the results of this solution are considered for the TREC Web graph, which is used to better illustrate the problem of One-Two Gap of the number of links between pages. We will conclude and summarize the discussion in the final section.

2. PageRank Algorithm

The PageRank algorithm works independently of the query, and it is used in the Google search engine. This algorithm runs on the entire web graph, and the rank of each page is equal to the total sum of the rank of its input pages; that is, a page with a high rank, with a large number of pages referring to, or pointed pages that have a high ranking [9], [10].

PageRank addresses the links between pages. For example, if the P_1 page has a connection to P_2, then the P_2 issue is probably interesting for the P_1 creator, so the number of links to the web pages indicates the degree of interest in the page for others. Clearly, the degree of interest in the page increases with increasing number of input links. Additionally, when the web page receives links from an important page, naturally it should have a higher ranking. PageRank of page j is displayed with r (j):

$$P_j = \frac{1-d}{n} + d \times \sum_{i \in B(j)} \frac{P_i}{O(i)} \tag{1}$$

Where O(i) represents the number of out-links from page i and B(j) represents the set of pages that refer to page j.

Therefore, PageRank j is equal to the total PageRank of the input pages divided by the degree of output. PageRank of the pages input divided into their out-degree O(i) has two effects. First, the distribution of PageRank to all outputs is fair; and secondly, the sum of the effect of each page and the vector of its page rank is normal. n is the total number of web pages in the web graph.

Parameter d is used to specify the probability of jumping to pages, which is in fact equivalent to random surfer behavior. When a user accesses a page without an out-link, it jumps to another page in random order; therefore, when a user is on a web page, with probability d, he chooses one of the random out-links or jumps to other pages with the probability of 1-d. Because this method is independent of the query, all pages compete with each other and reduce accuracy. This method suffers from a rich-get-richer problem [11]. In addition, the low utility coefficient of this algorithm is due to the lack of a web graph and the limited number of queries. The biggest advantage of PageRank is that it has nothing to do with the input (the query word), so all PageRank values are calculated as offline. It reduces online computing; however, the biggest defect in pagerank algorithm ignores the relevance of the subject with the information. Otherwise, Pages with different PageRanks can exist that have similar contents [12].

2.1 Overview of the Pagerank Problems

In this subsection, some of the similar tasks in the area of resolving PageRank problems are being examined to use existing ideas for further analysis of One-Two Gap problem.

In [13], three problems with using links are as follows:

- Two or more links may be created from a website or from two identical sites.
- Two or more links may be created from two similar sites to a site. In this case, two links should not be considered.
- Some links are created unrealistically for spam pages to raise their rank in search engines.

One of the PageRank problems is suspended pages [14]. Not all web pages have an out-link such as images, PDFs, and some explanatory pages and the like. Suspended pages are those that do not have an out link and they score points to their side like a hole.

A method was suggested for determining the spam linking of suspended pages. This method randomly selects a target page and identifies it by using a special vector and a special amount of spam; and then, by adding and removing the link will fix the problem. Eventually, the PageRank algorithm applies to the modified graph. The major problem with this method is its high execution time and the computational complexity that is practically impossible for large graphs [15].

In [16], a simple algorithm for calculating PageRank is presented. This algorithm considers all suspended webpages as a page and shows that the ranking of non-suspended webpages can be calculated independently of the pending page rank. Their performance has led to a ranking implemented on the smaller matrix. It was shown showed that the PageRank of suspended pages strongly affects non-suspended web pages, but it does not exist on the contrary. The benefits of this method are simple implementation and minimal storage.

In [17], Wang et al. raised the zero-one gap problem in PageRank. One method to calculate the privilege of suspended pages is to disconnect the inputs of these pages. In this method, the score obtained from the input pages is zero, and thus, there is a long difference between pages that do not have an out-link and those that have only one out-link. This problem is called the Zero-One Gap, and Wang et al. presented a new algorithm called DirchletRank to solve this problem. The DirchletRank algorithm is similar to the PageRank algorithm, with the difference that it does not have the Zero-One gap problem.

Bartlett et al. described another problem for graph-based methods [18]. They claimed that a link to a page could be a veto to determine the quality or lack of quality of that page. They described the problems raised in [13] in another way. For example, they identified links to spam pages that should be deleted in the polls or named repeated links that should be considered once in the voting. To solve these problems, they presented a new model called Super Graph. In the proposed model, the web super graph was constructed to categorize pages in distinct groups, and the main graph links were used to construct hypergraph links. In this way, the graph connections were more uniformly shaped.

Another problem of pagerank is the density in web graph [19]. Experiments show that the web graph is usually a Power Law distribution. In [19]-[25], experiments show that the distribution of PageRank, Out-degree and In-degree usually follow the Power Law distribution for different domains and different number of pages in the web graph. For example, the number of web pages with i inlinks is commensurate with $\frac{1}{i^{2.1}}$. This makes the connections matrix sparse matrix and thus scores assigned to many pages are more negligible and newborn pages receive a very small score.

Pang at el. [26] improved the PageRank algorithm by utilizing the content of the pages and time factor to resolve the problems of topic drift and emphasize older pages (the same problem of rich-get-richer). To reduce the rich-get-richer problem, Setayesh et al. created a new version of the PageRank algorithm that uses the interests of web page users and an ant colony algorithm [27].

The Norm-PageRank algorithm is a new version of PageRank. In each step, this algorithm normalizes web pages PageRank scores to the speed of convergence [28]. The TrustRank algorithm was presented by Google in 2005 [29] to reduce Link Spam. This algorithm considers trusted and well-known pages as the seed pages that do not leak into the spam page.

Xing et al. suggested the Weighted PageRank or in short WPR [30]. In their proposed algorithm, they received more weight, depending on their importance, instead of output pages of one page, which is received the same score from the previous page.

Another problem that is mentioned in [31] is the back button issue when it is redirected from a single page. In the PageRank algorithm, it is assumed that the user chooses one of the output links or jumps to another page, while the third mode is also possible and returns to the previous page, which is unlikely to be zero. Matthew and Bowellit [31] corrected the web graph by establishing a link between each page and the previous page.

When you log in to some pages, the page path will automatically be changed. This issue is also one of the challenges in the Web graph, and research has also been done in this area [32]. On the web, a group of pages may have links to each other and have no links out. This problem is called the spider-trap [33] and the method of removing it is similar to that of suspended pages.

In addition, in [34], a comparison was made between algorithms based on PageRank and methods that have remedied PageRank bugs. The DistanceRank algorithm, which is based on reinforcement learning, reduces one of the PageRank problems that is rich-get-richer [35].

Now, before expressing the One-Two Gap problem of the PageRank algorithm, the used terms are introduced in the next subsection.

2.2 Used Terms

Single node or single page: The nodes of the web graph, with a degree of output, are called single page. In fact, there is only one out-link on the pages equivalent to the single pages. The single page often relates to pages that are redirected automatically. Of course, in other cases, the single pages also appear. For example, some sites first describe their graphic problem and consider a button to opt cancel, which links that button to the main site; or some site designers add links to their sites at the bottom of the pages they are designing. Now, if the design page does not have a specific out link, it will appear as a single page in the graph due to the designer's link.

Single Chain: A redirection may be performed in several steps, and in practice, several single pages are referred to together; this set of single pages is called a single chain.

Length of the single chain: The number of edges that connect the pages in a single chain is called the length of the single chain. For example, in Fig. 2, there is only one single chain and its length is one, or in Fig. 3, the length of the single chain is two. In order to generalize this definition for pages that do not belong to any single chain, we consider the length of zero.

3. One-Two Gap Problem in PageRank

Here are a few examples of the One-Two Gap. Suppose Fig. 1 is a normal web graph with four web pages, with its PageRank values displayed next to the pages.

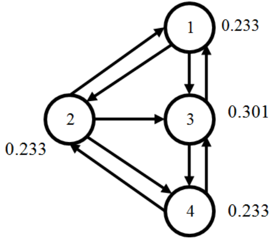


Fig. 1. Web graph without problems, pages suspended without problems one-two

If the graph in Fig. 1 adds the fifth page, that page 4 refers to and takes responsibility for page 4. PageRank scores will be in the form of Fig. 2.

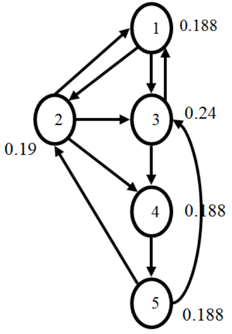


Fig. 2. Web graph with one-two Gap problem in page 4

As can be seen, page 4 has only one out-link and page 5, while having only 4 entries, has more points than page 4, which is not logical. More importantly, a new page that receives 1 entry from a page has a higher score than page 1.

If this trend continues and another page takes on the task of page 5, the scores credit will continue to decline. Fig. 3 shows this case. The new page has a score of page 6 over its equivalent pages.

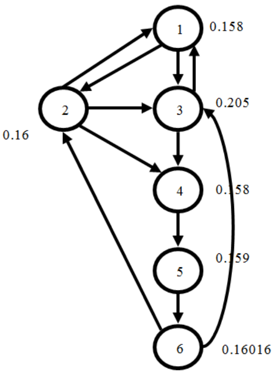


Fig. 3. Web graph with two One-Two Gap problem on Pages 4 and 5.

One way to make pages with out-degree is to use one of the ways to redirect the page to another page [32].

4. Identify Pages with One-Two Gap Problem

In this section, using a single lemma and a theorem, we identify the pages with the problem of One-Two Gap. With the tests performed, it became clear that this problem was not created for important pages, and that the nonsignificant pages had this problem. In Lemma (1), we show that nonsignificant pages with score PageRank are less than the inverse of the number of pages; and in theorem (1), we prove that the low-priority pages with the out degree of 1 have a problem of one-two Gap.

Lemma (1): If n is the total number of graph web pages, the less important pages with PageRank are less than 1/n [36].

Proof: In a normalized version, if the web graph has no pages suspended, the total PageRank scores will be 1 [36]; that is:

$$P_1 + P_2 + ... + P_n = 1 \qquad (2)$$

Now, if we assume that all web graph pages have a degree of importance, that is, all pi are equal to 1/n because we have:

$$n \times P_i = 1 \Rightarrow P_i = \frac{1}{n} \qquad (3)$$

So, if all web page graphs have a degree of importance, their PageRank score will be 1 / n, so more low importance pages with PageRank are less than 1/n.

Theorem (1): If a PageRank of a page with an out-degree of one is less than 1/n; in other words, the page is not important, the PageRank of the destination page is greater than the source page PageRank.

Proof: Fig. 4 Assuming page A with a PageRank out-degree of one and less than 1/n and page A to point page B.

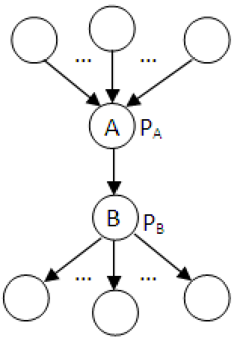


Fig. 4. Page A has an output degree of one

PageRank Score of page B will be:

$$P_B = \frac{1-d}{n} + d\sum_{i \in B_{in}} \frac{P_i}{O_i} \qquad (4)$$

$$P_B = \frac{1-d}{n} + d \times P_A \qquad (5)$$

$$P_B - P_A = \frac{1-d}{n} + d \times P_A - P_A \qquad (6)$$

$$P_B - P_A = (1-d) \times (1/n - P_A) \qquad (7)$$

Relation (4) is the PageRank formula used for page B. Only the sigma of relation (4) is the output of page A, according to this output degree of page A. P_A appears without a denominator in Sigma, and relation (5) is obtained. On the sides of equation (5), we reduce the P_A amount, and relation (6) is obtained, and then by factoring (1-d) relation (7) is obtained.

In relation (7), the expression (1-d) is always positive, and therefore the right-side term sign only on the sign (1 / n-P_A); in the event that P_A <1/n, P_B-P_A becomes greater than zero; or in other words P_B> P_A, the theorem is proved.

The gap may be diminished by increasing d, but with a large amount of d, it cannot be exploited by other links; therefore, the problem of One-Two gap is an inherent problem.

5. Proposed Model for Solving a One-Two Gap Problem

Perhaps the easiest way to solve the One-Two gap problem is to merge the pages with one output degree to the linked pages, but this solution may be the source of new problems. Firstly, the next pages may have other input links that are ambiguous with this change of status. Secondly, the two pages may not really fit together. For example, some site designers repeat the company's address as a link on each page, and if a page may have the same link, it should be merged into the company page that is not logical.

Another solution, was provided by Mathieu and Bouklit in [31]. They corrected the web graph by setting the link between each page and the previous page and added the role of the Back button to the graph. This method does not resolve one-two gap problem, and the out degree of suspended pages was one, and the same problem occurs for those pages.

The proposed solution is to prevent illogical publishing of scores from the page with the out degree of one to the next pages. PageRank scores are achieved in two ways. One is based on the score of the pages before it in the graph and one to jump to that page. The previous pages score is controlled by d, and since d is always less than one, it does not pass over the previous page's score to the desired page. The factor of increasing the score is the probability of jumping. If the two pages are identical, the probability of jumping is one, so the probability of jump cannot be assumed to be the same and should be halved. For example, the previous address of Yazd University was "www.yazduni.ac.ir" and has now been changed to "www.yazd.ac.ir" and the user can connect with one of two addresses to the Yazd site. Now, assigning two probabilities to a site causes a problem and somehow the whole web graph is affected; therefore, we should prevent from which is more than the source page score. To do this, we first rewrite the PageRank calculation formula as follows:

$$P_j = \sum_{i\in B(j)} \left(\frac{1-d}{n\times|B(j)|} + d\times\frac{P_i}{O(i)}\right) \tag{8}$$

In this relation, the variables are similar to Formula (1). With the difference that we differentiate for page j, in which part of the score is received from each in-link. In the stage, we can minimize the release of a score more than the source page score. For this purpose, the PageRank calculation formula is modified as follows:

$$P_j = \sum_{i\in B(j)} \min\left(P_i\,,\ \frac{1-d}{n\times|B(j)|} + d\times\frac{P_i}{O(i)}\right) \tag{9}$$

Formula (9) ensures that maximum released is P_i from page i to j, and this formula is a generalized formula for all degrees. Of course, for pages with zero entry, we still need to use formula (1).

In the following, using Lemma (2) and Theorem (2) we prove that (9) should not always be used, and in the calculation of PageRank, formula (1) can often be used.

Lemma (2): For pages such as page j, received input-link from page such as i only needs to use formula (9), which is $|B(j)| < \frac{1}{n\times P_i}$.

Proof: In the calculation of P_j when P_i is used as a minimum:

$$P_i < \frac{(1-d)}{n\times|B(j)|} + d\,\frac{P_i}{O(i)} \tag{10}$$

Due to the fact that most of the right side of inequality (10) occurs when the O(i) is equal to one, or, in other words, page i is single page; we have:

$$P_i \times (1-d) < \frac{(1-d)}{n\times|B(j)|} \tag{11}$$

$$|B(j)| < \frac{1}{n\times P_i} \tag{12}$$

And the lemma is proved.

Theorem (2): Equation (9) for a page like j only needs to be used when j is less than the input $\frac{1}{1-d}$.

Proof: According to equation (1), $\frac{1-d}{n}$ is the lowest Pi, so we have:

$$P_i > \frac{(1-d)}{n} \tag{13}$$

$$\frac{1}{1-d} > \frac{1}{n\times P_i} \tag{14}$$

By combining equation (14) with Lemma (2), we conclude that whenever the inputs of page j are to be checked, its input degree is less than the inverse of 1-d, that is:

$$|B(j)| < \frac{1}{1-d} \tag{15}$$

This makes it easier to process pages, and we use special cases of relation (9). Given that d is usually considered to be 0.85, [37], only for pages of less than or equal to six, Equation (9) is used.

In calculating score related to PageRank, scores are calculated recursive and destructive effects of pages with the out-degree of lower than six are transferred to the other pages as recursive.

This problem exists on these kinds of pages but in the moment of calculating the score of pages. Because of the reclusiveness of the calculation and the score of these pages are effected in the scores of pages which have a link between them as a chain-by-links. So the wider range of pages will be affected by the problem. For example, if a page has the One-Two Gap problem and is linked in series to a page with the highest out-degree, the calculation of this score also causes an error.

In this paper, we tried to solve the problem with the least computations and solve the problem of these pages in order to not solve the problem in the whole graph.

The pseudocode of proposed solution is shown in algorithm 1. In Algorithm 1, the PageRank formula based on the previous discussion is used.

Algorithm 1: PageRank algorithm without One-Two Gap problem

```
procedure PageRank_Without_One-Two_Gap($G$, $iteration$) %$G$: inlink file, $iteration$: # of iteration
    $d \leftarrow 0.85$                                   %damping factor: 0.85
    $oh \leftarrow G$                                     %get outlink count hash from $G$
    $ih \leftarrow G$                                     %get inlink hash from $G$
    $N \leftarrow G$                                      %get # of pages from $G$
    for all $p$ in the graph do
        $opg[p] \leftarrow \frac{1}{N}$        %/initialize PageRank_Without_One-Two_Gap
    end for
    while $iteration > 0$ do
        for all $p$ in the graph do
            for all $ip$ in $ih[p]$ do
                if $|ih[p]| <= 6$ then
                    $npg[p] \leftarrow npg[p] + \min(opg[ip],\ \frac{1-d}{N*|ih[p]|} + \frac{d*opg[ip]}{oh[ip]})$
                else
                    $npg[p] \leftarrow npg[p] +$
```

```
$(\frac{1-d}{N*|ih[p]|} + \frac{d*opg[ip\,]}{oh[ip]})$
%get PageRank_Without_One-Two_Gap from inlinks and get PageRank_Without_One-Two_Gap from random jump
                end if
            end for
        end for
    $opg \leftarrow npg$ %update PageRank_Without_One-Two_Gap
    $iteration \leftarrow iteration - 1$
    end while
end procedure
```

Fig. 5 shows a modified PageRank algorithm in which relation (1) based on relation (9) is modified.

5.1 Case Study

With an example on a graph with ten pages and 17 edges, we simulate the proposed solution, as shown in Fig. 5. Therefore, we can check the validity of the fixed One-Two gap problem. The simulation is done in the Matlab environment and tested on the Intel core 7 system with a six GB RAM. The proximity matrix A and the transition matrix P graph are created as follows.

$$A = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \text{ and}$$

$$P = \begin{bmatrix} 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 1 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \end{bmatrix}$$

For the PageRank algorithm and the proposed solution, the damping factor d is 0.85 and the threshold of 0.000000001 for the smaller squared error condition is set to be two consecutive repetitions. Fig. 5 shows that pages 1, 5, and 6 contain the problem of One-Two gap. The two left-hand figures have been used in different colors and sizes to show different rankings. This is also enhanced when the fixed problem is also more diverse rank. Two charts on the right side, from top-to-down, show the ranking pages of the pages before and after the One-Two gap problem.

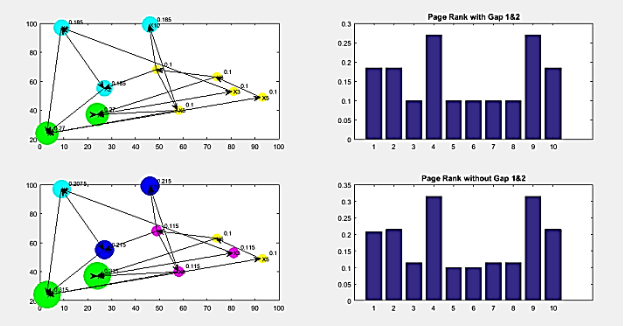


Fig. 5. The output of the page rank and their graph for the algorithms of PageRank and PageRank without One-Two gap problem

6. Experiments

We are using a TREC[1] standard data set in the .GOV domain, which was crawled in 2002. This dataset contains 50 queries, and for each query, the web pages linked to it are identified. The TREC size is about 18 gigabytes and includes 1247753 crawled web pages in the .GOV domain [38]. In this series of experiments, a number of links between pages were removed to contain 39,635 pages, including the terms of theorem (2) in the entire graph of the web, so that there is a possibility to check the validity of the proposed solution.

In this article, some links between pages have been removed in order to increase the damage effect of The One-Two Gap problem on PageRank privileges of many the pages in the web graph. And also because it is evaluated precision in the first ten ranks. Therefore, it is necessary to ensure that there is at least a page contains the One-Two Gap problem on the web graph that is relevant or linked to the relevant pages in the form of link chain. (Note that it is checked that the power law distribution of the Web graph is not lost by deleting the links).

6.1 Experiment 1: Convergence Review

After changing the PageRank algorithm to solve the One-Two gap problem, the first test is to test the method convergence to see if the algorithm's accuracy has not been lost. To prove the empirical convergence of the proposed solution, a similarity test is performed. To illustrate convergence, the results of the repetitions are compared with the last one. For this reason, we obtained the similarities of the repetitions of 10, 20, 30, 40, 50, 60, 70, 80 and 90 with 100th repetition. The similarity of the two lists is calculated according to the following equation [39].

$$\text{Similarity} = \frac{|A \cap B|}{|A \cup B|} \qquad (16)$$

Where A and B represent a list of related pages from different iterations. $|A \cup B|$ indicates the total number of pages that have two lists (Union of two lists) and $|A \cap B|$ indicates the number of pages that appear on both lists (Subscribe to two lists). To draw a chart, the list of web pages is based on the N page of the sorted list, which is the

[1] http://trec.nist.gov/

horizontal axis of the graph. When the similarity of the two lists from each of the iterations is close to one, it means that the list of pages is unchanged and convergence is complete.

Fig. 6 shows the convergence of the PageRank algorithm without One-Two gap problem. The algorithm is very close to the one after the 50th iteration.

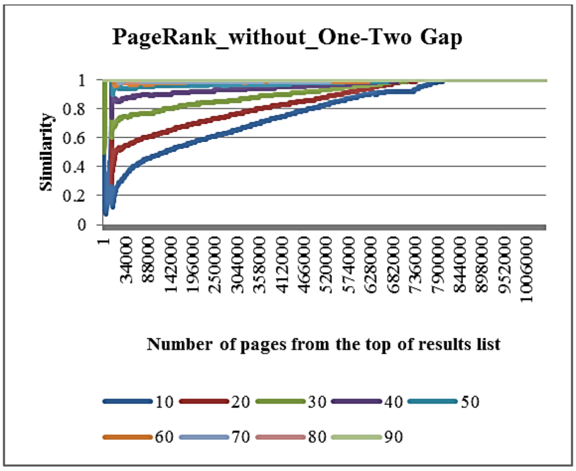


Fig. 6. The convergence of the PageRank algorithm without One-Two gap problem by comparing the similarity of repetitions with 100th repetition

Table 1. The convergence rate of the PageRank algorithm and PageRank without one-two gap problem using a 100-repeat similarity criterion

Start round number of Convergence Over 90%	Start round number of Convergence 100%	Algorithm
50	90	PageRank
50	90	PageRank without One-Two GAP

The results of Table 1 show that the convergence rate is not reduced by applying the change. In both cases, the problem of the One-Two gap and without this problem are obtained in repeats of 50 and 90, respectively, with the convergence of over 90% and complete convergence.

6.2 Experiment 2: Reviewing the Percentage Demoted of the PageRank Algorithm without the One-Two Gap Problem Compared to the PageRank Algorithm

We first consider a relative ranking change (RRC) measurement for web pages. Suppose that page i in position s in PageRank has no gap of One-Two and in position t of PageRank algorithm. The RRC criterion is defined as follows.

$$\mathrm{RRC(i)} = \frac{\mathrm{s-t}}{\mathrm{s+t}} \qquad (17)$$

The RRC (i) is in the range [-1,1]. The positive RRC (i) indicates an increase in rank of page i and an RRC (i) negatively indicating a downgrade of the rank of page i. The RRC is relative to the high-ranking position. We consider the percentage Demoted (PD) based on the RRC for the TREC web graph pages, which is calculated as follows [17].

$$\mathrm{PD(s)} = \frac{\left|\sum_{\mathrm{i\epsilon s,RRC(i)>0}} \mathrm{RRC(i)}\right|}{\left|\sum_{\mathrm{i\epsilon s,RRC(i)<0}} \mathrm{RRC(i)}\right| + \left|\sum_{\mathrm{i\epsilon s,RRC(i)>0}} \mathrm{RRC(i)}\right|} \qquad (18)$$

The PD value represents the percentage demoted of pages for RRC. We use without the One-Two gap problem to show the effectiveness of PageRank, which is about 52% more demoted pages than PageRank.

Fig. 7 shows the percentage demoted ranking of pages in the pagerank algorithm without One-Two gap problem. The demoted rank of algorithm is 52%.

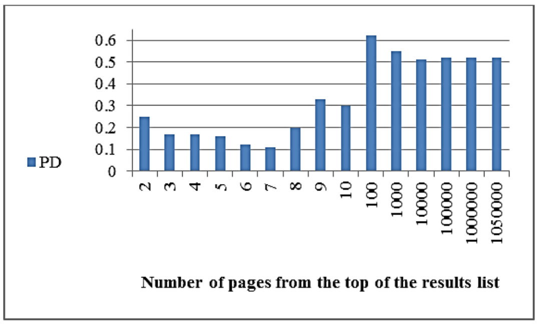


Fig. 7. Percentage demoted of the PageRank algorithm without One-Two gap problem relative to the PageRank algorithm in the different number of pages

6.3 Test 3: Evaluation the Accuracy of the Ranking

The ranking accuracy was performed using the three criteria of P@n, NDCG@n, MAP evaluation on the TREC graph, which removed a series of links between the pages.

6.3.1 Precision Evaluation Criteria

In the retrieval of information, precision and recall are used as criteria for checking the efficiency and quality of the ranking [39]. Precision criteria are used in the position n-th (p@n), mean-average precision (MAP), Normalized Discount Cumulative Gain (NDCG), and recall n-th (R@n) to evaluate the accuracy of information retrieval. The evaluation tools set of LETOR group supports these criteria [38].

- P@n

This criterion indicates the number of relevance pages to user's query in the n position of the ranking list. Relation P@n is as follows.

$$\mathrm{P@n} = \frac{\mathrm{NoR_n}}{\mathrm{n}} \qquad (19)$$

Where $\mathrm{NoR_n}$ indicates the number of relevance pages in the n position of top ranking list.

- MAP

The Mean Average Precision (MAP) represents the average AP values for all queries provided, and for each query, the AP indicates average of P@n values for all relevance pages.

$$\mathrm{AP} = \frac{\sum_{\mathrm{n=1}}^{\mathrm{N}} (\mathrm{P@n} \cdot \mathrm{R_e(n)})}{\mathrm{T_R}} \qquad (20)$$

Where N, $\mathrm{T_R}$ and $\mathrm{R_e(n)}$ are the number of retrieved pages, the number of relevance pages, and the binary function of the n-th page, respectively, indicating the relevance page with one and the irrelevance page with zero.

- NDCG

The NDCG value of a ranked page in the n-th position is computed as follows:

$$\mathrm{NDCG} = \mathrm{Z_n} \sum_{i=1}^{N} (2^{r(i)} - 1/\log(i+1)) \qquad (21)$$

Where $\mathrm{Z_n}$ represents normalization constant and $r(i)$ indicates the relevance level of page i in ranking list. The gain of the i-th page and the discount gain are calculated with the relations $2^{r(m)} - 1$ and $2^{r(i)} - 1/\log(i+1)$. $\sum_{i=1}^{N} (2^{r(i)} - 1/\log(i+1))$ represents the normalized discount cumulative gain in the n-th position.

- R@n

Recall indicates the proportion of retrieved pages that are relevant to the query, and it is called sensitivity [40]. R@n is calculated as follows.

$$\mathrm{R@n} = \frac{\mathrm{NoR_n}}{\mathrm{NoR\_all}} \qquad (22)$$

Where $\mathrm{NoR_n}$ indicates the number of relevance pages in the top-n result of ranking list. *NoR_all* shows the whole number of relevance pages to the query.

Regarding Figs. 8-10, the proposed solution, compared to the PageRank algorithm was more appropriate in terms of the P@n, MAP, and NDCG@n evaluation criteria on the TREC2003 benchmark dataset. Due to the fact that the dataset has the conditions for the One-Two gap problem, according to the results, the proposed solution to solve the One-Two gap problem will work perfectly.

According to the two criteria, P@n and NDCG@n, for the first ten pages of the ranking list and MAP criteria, ranking accuracy has been enhanced by solving the problem of One-Two gap.

This precision, according to the P@n criterion in the position of one and two from ranking list, increased by 50% and 100%, respectively. According to the MAP criterion, the overall precision has increased by about 0.04%.

Fig. 11 shows the sensitivity of proposed solution better than the PageRank algorithm. In this figure, it is seen that in the top-10 rank of the ranking list, the PageRank without One-Two gap problem is more recall, and the rhythm of both methods is incremental. It represents that performance of the proposed solving way is much better than PageRank.

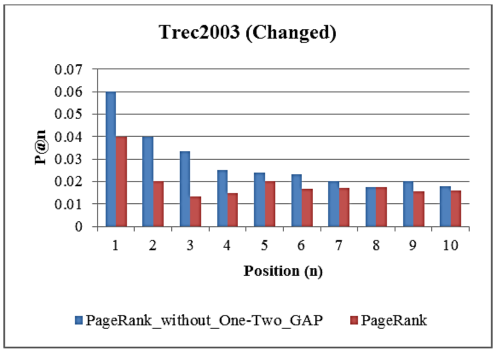


Fig. 8. Comparing the proposed solution with the PageRank algorithm on the TREC2003 data set based on the P@n criterion

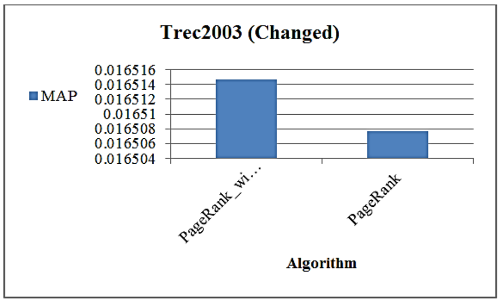


Fig. 9. Comparing the proposed solution with the PageRank algorithm on the TREC2003 data set based on the MAP criterion

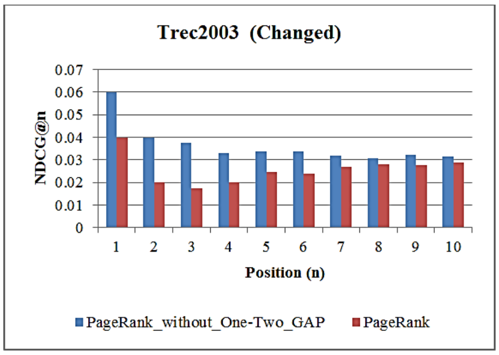


Fig. 10. Comparing the proposed solution with the PageRank algorithm on the TREC2003 data set based on the NDCG@n criterion

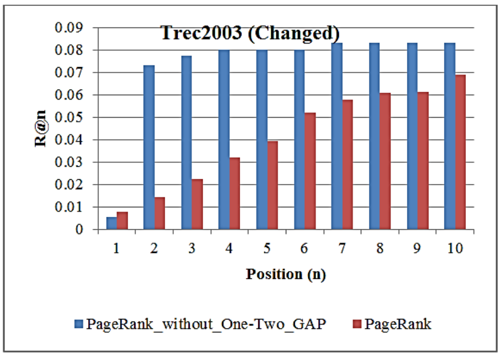


Fig. 11. Comparing the proposed solution with the PageRank algorithm on the TREC2003 data set based on the R@n criterion

6.4 Discussion and Analysis

We have theoretically shown that there is a One-Two gap problem in the PageRank algorithm. The proposed solution is a prerequisites for the One-Two gap problem. Not all the methods that have been developed based on PageRank have paid attention to this problem. In general, the proposed solution provides the right or equal value for PageRank, and can be a good alternative to this algorithm.

Experimental results also emphasize the appropriateness of the proposed solution's performance in terms of accuracy, sensitivity and convergence. The proposed method offers a good combination of sensitivity and accuracy in the top-10 rank of results. Further, the results reveal that this solution contributes to achieving better precision and recall.

The findings also establish the case that the One-Two gap problem decreases precision, accuracy, and recall in PageRank, and this article solves it. In other word, this method prevents the release of the wrong score to other pages in whole graph.

7. Conclusion

While the PageRank algorithm is used successfully in Google's search engine, there are many researchers who pay attention to it, and many advanced methods have been proposed to improve the precision of this algorithm. In addition, the PageRank algorithm is considered one of the factors that calculates the relevance of web pages. In this paper, we have shown that link-based PageRank algorithm has the One-Two gap problem, which can put the rank of web page more than linked page; and ranks are calculated to be recursive to make errors. We have suggested a solution to this problem, which has been empirically increasing the precision of the ranking. In terms of convergence, the proposed solution is similar to the PageRank algorithm. PageRank without One-Two gap problem acquired the highest recall and precision than PageRank algorithm, on the TREC2003 dataset in domain .gov.

References

[1] K. Purcell, J. Brenner, and L. Rainie, “Search Engine Use 2012,” 2012.

[2] Y. Zhang, B. J. Jansen, and A. Spink, “Time series analysis of a Web search engine transaction log,” Information Processing & Management, vol. 45, no. 2, pp. 230–245, 2009.

[3] R. Baeza-Yates and B. Ribeiro-Neto, “Modern information retrieval,” New York, vol. 9, p. 513, 1999.

[4] Searchmetrics, “Searchmetrics Ranking Factors 2016: Rebooting for Relevance,” 2016. [Online]. Available: http://www.searchmetrics.com/knowledge-base/ranking-factors/. [Accessed: 07-May-2017].

[5] L. Page, S. Brin, R. Motwani, and T. Winograd, “The PageRank Citation Ranking: Bringing Order to the Web,” World Wide Web Internet And Web Information Systems, vol. 54, no. 1999–66, pp. 1–17, 1998.

[6] G.-R. Xue, Q. Yang, H.-J. Zeng, Y. Yu, and Z. Chen, “Exploiting the hierarchical structure for link analysis,” Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 186--193, 2005.

[7] J. M. Kleinberg, “Authoritative sources in a hyperlinked environment,” Journal of the ACM, vol. 46, no. 5, pp. 604–632, Sep. 1999.

[8] R. Patchmuthu, “Link analysis algorithms to handle hanging and spam pages,” 2014.

[9] L. Page, S. Brin, R. Motwani, and T. Winograd, “The PageRank Citation Rankin: Bringing Order to the Web,” World Wide Web Internet And Web Information Systems, vol. 54, no. 1999–66, pp. 1–17, 1998.

[10] M. Bianchini, M. Gori, and F. Scarselli, “Inside pagerank,” ACM Transactions on Internet Technology (TOIT), vol. 5, no. 1, pp. 92–128, 2005.

[11] A.-L. Barabási and R. Albert, “Emergence of Scaling in Random Networks,” Science, vol. 286, no. October, pp. 509–512, 1999.

[12] L. Z. Xiang, “Research and Improvement of PageRank Sort Algorithm Based on Retrieval Results,” in Intelligent Computation Technology and Automation (ICICTA), 2014 7th International Conference on, 2014, pp. 468–471.

[13] Z. Dou, R. Song, J.-Y. Nie, and J.-R. Wen, “Using anchor texts with their hyperlink structure for web search,” in Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, 2009, pp. 227–234.

[14] A. N. Langville and C. D. Meyer, “A reordering for the PageRank problem,” SIAM Journal on Scientific Computing, vol. 27, no. 6, pp. 2112–2120, 2006.

[15] R. K. Patchmuthu, A. K. SINGH, and A. Mohan, “A new algorithm for detection of link spam contributed by zero-out link pages,” Turkish Journal of Electrical Engineering & Computer Sciences, vol. 24, no. 4, pp. 2106–2123, 2016.

[16] I. C. F. Ipsen and T. M. Selee, “PageRank computation, with special attention to dangling nodes,” SIAM Journal on Matrix Analysis and Applications, vol. 29, no. 4, pp. 1281–1296, 2007.

[17] X. Wang, T. Tao, J.-T. Sun, A. Shakery, and C. Zhai, “Dirichletrank: Solving the zero-one gap problem of pagerank,” ACM Transactions on Information Systems (TOIS), vol. 26, no. 2, p. 10, 2008.

[18] K. Berlt, E. S. de Moura, A. Carvalho, M. Cristo, N. Ziviani, and T. Couto, “Modeling the web as a hypergraph to compute page reputation,” Information Systems, vol. 35, no. 5, pp. 530–543, 2010.

[19] A. N. Nikolakopoulos and J. D. Garofalakis, “NCDawareRank: a novel ranking method that exploits the decomposable structure of the web,” in Proceedings of the sixth ACM international conference on Web search and data mining, 2013, pp. 143–152.

[20] A. Broder et al., “Graph structure in the web,” Computer networks, vol. 33, no. 1, pp. 309–320, 2000.

[21] D. Donato, L. Laura, S. Leonardi, and S. Millozzi, “Large scale properties of the webgraph,” The European Physical Journal B-Condensed Matter and Complex Systems, vol. 38, no. 2, pp. 239–243, 2004.

[22] A. Flammini, F. Menczer, and A. Vespignani, “The egalitarian effect of search engines,” arXiv preprint cs.CY/0511005, 2005.

[23] L. Becchetti and C. Castillo, “The distribution of PageRank follows a power-law only for particular values of the damping factor,” in Proceedings of the 15th international conference on World Wide Web, 2006, pp. 941–942.

[24] G. Pandurangan, P. Raghavan, and E. Upfal, “Using pagerank to characterize web structure,” Internet Mathematics, vol. 3, no. 1, pp. 1–20, 2006.

[25] N. Litvak, W. R. W. Scheinhardt, and Y. Volkovich, “In-Degree and PageRank of Web pages: Why do they follow similar power laws?,” arXiv preprint math/0607507, 2006.

[26] P. Zha, X. Xu, and M. Zuo, “An Efficient Improved Strategy for the PageRank Algorithm,” in 2011

International Conference on Management and Service Science, 2011, pp. 1–4.

[27] S. Setayesh, A. Harounabadi, and A. M. Rahmani, "Presentation of an Extended Version of the PageRank Algorithm to Rank Web Pages Inspired by Ant Colony Algorithm," International Journal of Computer Applications, vol. 85, no. 17, 2014.

[28] K. Mohan and J. Kurmi, "A Technique to Improved Page Rank Algorithm in perspective to Optimized Normalization Technique," International Journal, vol. 8, no. 3, 2017.

[29] N. L. Amy and D. M. Carl, "Google's PageRank: The Math Behind the Search Engine," Priceton university Press, Nol, vol. 3, pp. 335–380, 2004.

[30] W. Xing and A. Ghorbani, "Weighted PageRank algorithm," in Proceedings of the Second Annual Conference on Communication Networks and Services Research, 2004, pp. 305–314.

[31] F. Mathieu and M. Bouklit, "The effect of the back button in a random walk: application for pagerank," in Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters, 2004, pp. 370–371.

[32] M. Zhukovskii, G. Gusev, and P. Serdyukov, "URL redirection accounting for improving link-based ranking methods," in Advances in Information Retrieval, Springer, 2013, pp. 656–667.

[33] Z. Bahrami Bidoni, R. George, and K. Shujaee, "A Generalization of the PageRank Algorithm," ICDS 2014, The Eighth International Conference on Digital Society, pp. 108–113, 2014.

[34] A. K. Singh, "A comparative study of page ranking algorithms for information retrieval," International journal of electrical and computer engineering, vol. 4, pp. 469--480, 2009.

[35] A. M. Zareh Bidoki and N. Yazdani, "DistanceRank: An intelligent ranking algorithm for web pages," Information Processing and Management, vol. 44, no. 2, pp. 877–892, 2008.

[36] B. Poblete, C. Castillo, and A. Gionis, "Dr. searcher and mr. browser: a unified hyperlink-click graph," in Proceedings of the 17th ACM conference on Information and knowledge management, 2008, pp. 1123–1132.

[37] R. Baeza-Yates, P. Boldi, and C. Castillo, "Generalizing pagerank: Damping functions for link-based ranking algorithms," in Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, 2006, pp. 308–315.

[38] T. Qin, T. Y. Liu, J. Xu, and H. Li, "LETOR: A benchmark collection for research on learning to rank for information retrieval," Information Retrieval, vol. 13, pp. 346–374, 2010.

[39] T. H. Haveliwala, "Efficient computation of PageRank," 1999.

[40] M. Arora, U. Kanjilal, and D. Varshney, "Evaluation of information retrieval: Precision and recall," International Journal of Indian Culture and Business Management, vol. 12, no. 2, pp. 224–236, 2016.

Javad Paksima received the B.Sc. degree in Software engineering from Sharif University, Tehran, Iran, in 1996. He received M.Sc. degree in Software engineering from Sharif University, Tehran, Iran, in 1998. He received Ph.D. degree in Software engineering from Yazd University, Yazd, Iran, in 2018. He is a faculty member of Payam Noor University (PNU). His research interests include Search engines, Algorithms design and Parallel Programming.

Homa Khajeh received the B.Sc. degree in Software Engineering from Islamic Azad University, Najafabad Branch (IAUN), in Isfahan, Iran, in 2009 and her M.Sc. degree of Software Engineering from Science and Art University in Yazd, Iran, in 2014. Her research interests are mainly in the field of Information Retrieval, Search Engine, Machine Learning, and Big Data.