NORMALIZATION OF DUPLICATE RECORDS FROM MULTIPLE SOURCES

CHIKKAM MADHURI
PG Scholar, Department of M.C.A,
S.K.B.R P.G College,
Amalapuram, E.G.Dt., A.P, India.
chmadhu367@gmail.com

Mr. NAGA. SRINIVASA RAO*
Asst. Professor, Dept of M.C.A,
S.K.B.R P.G College,
Amalapuram, E.G.Dt., A.P, India.
naagaasrinu@gmail.com

Abstract:

Data consolidation is a challenging issue in data integration. The usefulness of data increases when it is linked and fused with other data from numerous (Web) sources. The promise of Big Data hinges upon addressing several big data integration challenges, such as record linkage at scale, real-time data fusion, and integrating Deep Web. Although much work has been conducted on these problems, there is limited work on creating a uniform, standard record from a group of records corresponding to the same real-world entity. We refer to this task as record normalization. Such a record representation, coined normalized record, is important for both front-end and back-end applications. In this paper, we formalize the record normalization problem, present in-depth analysis of normalization granularity levels (e.g., record, field, and value-component) and of normalization forms (e.g., typical versus complete). We propose a comprehensive framework for computing the normalized record. The proposed framework includes a suit of record normalization methods, from naive ones, which use only the information gathered from records themselves, to complex strategies, which globally mine a group of duplicate records before selecting a value for an attribute of a normalized record. We conducted extensive empirical studies with all the proposed methods. We indicate the weaknesses and strengths of each of them and recommend the ones to be used in practice.

Keywords : Data integration, Standards, Task analysis, Databases, Google, Data mining, Terminology

1.INTRODUCTION

1.1 Introduction:

The usefulness of Web data increases exponentially (e.g., building knowledge bases, Web-scale data analytics) when it is linked across numerous sources. Structured data on the Web resides in Web databases and Web tables. Web data integration is an important component of many applications collecting data from Web databases, such as Web data warehousing (e.g., Google and Bing Shopping; Google Scholar), data aggregation (e.g., product and service reviews), and met searching.

Integration systems at Web scale need to automatically match records from different sources that refer to the same real-world entity find the true matching records among them and turn this set of records into a standard record for the consumption of users or other applications. There is a large body of work on the record matching problem and the truth discovery problem. The record matching problem is also referred to as duplicate record detection, record linkage, object identification, entity resolution, or de-duplication and the truth discovery problem is also called as truth finding or fact finding - a key problem in data fusion.

This work assumes that the tasks of record matching and truth discovery have been performed and that the groups of true matching records have thus been identified. Our goal is to generate a uniform, standard record for each group of true matching records for end-user consumption. It calls the generated record the normalized record. It call the problem of computing the normalized record for a group of matching records the record normalization problem (RNP), and it is the focus of this work.

RNP is another specific interesting problem in data fusion. Record normalization is important in many application domains. For example, in the research publication domain, although the integrator website, such as Citeseer or Google Scholar, contains records gathered from a variety of sources using automated extraction techniques, it must display a normalized record to users. Otherwise, it is unclear what can be presented to users: (i) present the entire group of matching records or (ii) simply present some random record from the group, to just name a couple of ad-hoc approaches. Either of these choices can lead to a frustrating experience for a user, because in (i) the user needs to sort/browse through a potentially large number of duplicate records, and in (ii) it run the risk of

presenting a record with missing or incorrect pieces of data. Record normalization is a challenging problem because different Web sources may represent the attribute values of an entity in different ways or even provide conflicting data. Conflicting data may occur because of incomplete data, different data representations, missing attribute values, and even erroneous data.

This work aims to develop a framework for constructing normalized records systematically. This work includes a suit of record normalization methods, from naive ones, which use only the information gathered from records themselves, to complex strategies, which globally mine a group of duplicate records before selecting a value for an attribute of a normalized record.

1.2 Purpose:

Record normalization is a challenging problem because different Web sources may represent the attribute values of an entity in different ways or even provide conflicting data. Conflicting data may occur because of incomplete data, different data representations, missing attribute values, and even erroneous data. For example, Table 1 contains four records corresponding to the same entity (publication). They are extracted from different websites. Record Rnorm is constructed by hand for illustration purposes. One notices that the same publication has different representations in different websites.

1.3 Scope:

Integration systems at Web scale need to automatically match records from different sources that refer to the same real-world entity find the true matching records among them and turn this set of records into a standard record for the consumption of users or other applications. There is a large body of work on the record matching problem and the truth discovery problem. The record matching problem is also referred to as duplicate record detection, record linkage , object identification, entity resolution ,or deduplication and the truth discovery problem is also called as discovery have been performed and that the groups of true matching records have thus been identified. Our goal is to generate a uniform, standard record for each group of true matching records for end-user consumption. We call the generated record the normalized record. We call the problem of computing the normalized record for a group of matching records the record normalization problem (RNP), and it is the focus of this work. RNP is another specific interesting problem in data fusion.

1.4 Motivation:

Record normalization is important in many application domains. For example, in the research publication domain, although the integrator website, such as Citeseer or Google Scholar, contains records gathered from a variety of sources using automated extraction techniques, it must display a normalized record to users. Otherwise, it is unclear what can be presented to users: (i) present the entire group of matching records or(ii)simply present some random record from the group, to just name a couple of ad-hoc approaches. Either of these choices can lead to a frustrating experience for a user, because in (i) the user needs to sort/browse through a potentially large number of duplicate records, and in (ii) we run the risk of presenting a record with missing or incorrect pieces of data.

1.5 Overview:

We identify three levels of normalization granularity: record, field, and value-component. Record level assumes that the values of the fields within a record are governed by some hidden criterion and that together create a cohesive unit that is user-friendly. As a consequence, this normalization favors building the normalized record from entire records among the set of matching records rather than piecing it together from field values of different records.Thus,any of the matching records(ideally, that has no missing values) can be the normalized record. Using our running example in Table 1, the record Rc is a possible choice for the normalized record with this level of normalization granularity. Field level assumes that record level is often inadequate in practice because records contain fields with incomplete values. Recall that these records are the products of automatic data extraction tools, which are not perfect and thus may produce errors [18]. This normalization level ignores the cohesion factor in the record normalization level and assumes that a user is better served when each field of the normalized record has as easy to understand a value as possible, selected from among the values in the set of matching records.

2. RELATED WORK

SanghyeonBaeg [1] 2008, Power consumption is the most critical issue for low-power ternary content-addressable memory (TCAM) in match lines designs. In the proposed match-line architecture, the match line present in each TCAM word is partitioned into four segments and is selectively pre-charged to reduce the match-line power consumption. The match lines which are partially charged are evaluated to

determine the final comparison result by sharing the charges deposited in various parts of the partitioned segments.

B. Heller et al, [2] 2010, Built ElasticTree, which through data-center-wide traffic management and control, introduces energy proportionality in today's non-energy proportional networks. They will likely essentially decrease this quickly developing vitality cost. Compare multiple strategies for finding the minimum-power network [20]. The framework is vitality proficiency, best execution, and adaptation to non-critical failure. The system worked near its ability will build the possibility of dropped and postponed bundles.

A.R. Curtis et al, [3] 2011, DevoFlow proposition enables administrators to target just the streams that issue for their administration issue. DevoFlow handles most miniaturized scale streams in the information plane and consequently enables us to make the most out of switch resources. DevoFlow takes care of the issue by permitting a clonabletrump card principle to choose a yield port. Multipath steering to statically stack balance movement with no utilization of the control-plane. These procedures don't spare much vitality on elite systems.

P. Porraset al, [4] 2012, Incorporates several critical components that are necessary for enabling security applications in Open Flow networks including role-based authorization, rule reduction, conflict evaluation, and policy synchronization. FortNOX is a critical initial phase in enhancing the security of Open Flow systems. It shows the achievability and suitability of our nom de plume set guideline decrease approach [18]. It is unable to handle the dynamic matching process.

Zahid Ullah et al, [5] 2012, Hybrid partitioned static random is a memory architecture in which access memory-based ternary content addressable memory (HP SRAM-based TCAM), which involves TCAM functionality with conventional SRAM, where we are eliminating the inherited disadvantages of conventional TCAMs. HP SRAM-based TCAM is a technique in which they logically dissect conventional TCAM table in a hybrid way (column-wise and row-wise) into TCAM sub-tables, which are then processed to be mapped to their corresponding SRAM memory units.

H. Kim and N. Feamster et al, [6] 2013, Designed and implemented Procera, an event-driven network control framework

based on SDN. Additionally, utilize the OpenFlow convention to impart between the Procera controller and the hidden system switches. It gives better permeability and command over undertakings for performing system. This SDN can improve common network management tasks [19]. Procera experiences the characteristic deferral caused by the communication of the control plane and the information plane.

M. Yu, L. Jose et al, [7] 2013, OpenSketch empowers a straightforward and proficient approach to gather estimation information. It utilizes information plane estimation natives dependent on ware switches and an adaptable control plane so administrators can without much of a stretch execute variable estimation calculations. It has a simple, efficient way to control switches [16]. Sketches more flexible in supporting various measurement tasks. Delay of each measurement pipeline component is large.

Weirong Jiang et al, [8] 2013, Random access memory i.e. (RAM)-based Ternary Content Addressable Memory i.e.(TCAM) architecture is design for efficient implementation on state-of-the-art FPGAs. We give a formal study on RAM-based TCAM to disclose the ideas and the algorithms behind it. To face the timing challenge, we propose a modular architecture consisting of arrays of small-size RAM-based TCAM units.

Jacobson et al, [9] 2014, Novel control plane architecture called OpenNF that addresses these challenges through careful API design. OpenNF enables applications to settle on reasonable decisions in meeting their destinations. NF software is always Up-to-Date. The system has High performance on network monitoring.

M. Moshref et al, [10] 2014, DREAM enables operators and cloud tenants to flexibly specify their measurement tasks in a network and dynamically allocates TCAM resources to these tasks based on the resource-accuracy. User-specified high level of accuracy. DREAM can support more concurrent tasks. DREAM needs to dismiss almost half of the assignments and drop about 10%.

N. Katta et al, [11] 2014, CacheFlow system is a system which “caches” the most popular rules in the small TCAM, in which they are relying on software to handle the small amount of “cache miss” traffic. But, we cannot blindly apply existing cache-replacement algorithms, because of

dependencies between rules with overlapping patterns.

Naga Katta et al, [12] 2014, Instead of creating long dependency chains to cache smaller groups of rules in which semantics of the network policy are preserved. There are mainly four types of criteria for it. Elasticity which combines the best of hardware and software switches. Transparency which faith-fully supporting native OpenFlow semantics, including traffic counters. Fine-grained rule caching which places popular rules in the TCAM, despite dependencies on less-popular rules. Adaptability which enables incremental changes to the rule caching as the policy changes.

3. EXISTING SYSTEM

In existing, TCAM Razor, DomainFlow and Palette algorithms are used. Which is both power hungry and highly limited in capacity . Most TCAM-capable commodity switches support only a few thousand wildcard entries. Although certain products recently reported an ability to support up to 125k wildcard entries, enlarging the memory with enhanced control capability significantly increases the cost To improve scalability, two approaches have been taken: proactively allocating rules on multiple switches to load balance the memory consumption , and reactively caching rules on each switch individually.

3.1 Disadvantages: These existing works provides poor caching ratio and less hit ratioThese existing works provides poor caching ratio and less hit ratio

4. PROPOSED SYSTEM

To deal with existing disadvantages, this work proposed a novel wildcard-rule caching algorithm and a cache replacement algorithm to make use of TCAM space efficiently.TCAM can look up a packet's header and compare the matching patterns of the packet to the match field of all rules in the flow table in parallel. Our wildcard-rule caching algorithm repeats caching a set of important rules into TCAM until there is no TCAM space. Our cache replacement algorithm takes temporal and spatial traffic localities into consideration, which could make hit ratio high.

4.1 Advantages

The proposed wildcard-rule caching algorithm could have better caching ability than the other existing algorithms. Furthermore, the proposed cache replacement algorithm could have higher hit ratio than the other existing algorithms.

5. IMPLEMENTATION

5.1 Load conference name dataset:

This module load conference name dataset.This dataset contain rid, label and conference name.This dataset contains 3683 records.

5.2 Mining Abbreviation Definition pairs:

This module use a number of heuristics to determine whether given two value components s and t, s is an abbreviation of t. In this section, a value component is a word (or term). As we mentioned previously, in this module we consider only fields with the string data type. We define the neighboring context of a word w within the set of values of a field fj as the set of pairs (left neighbor word, right neighbor word) with the property that the substring left neighbor word w right neighbor word is a substring of a value of fj in some record in Re. If w is the beginning word of a field value, we use a special start-symbol "<s>" to mark left neighbor word. If it is the last word in the field value, we use the special end-symbol "</s>" to mark right neighbor word. For example, the words "proceedings" and "proc" occur many times in the field venue, and they share a good fraction of their neighboring contexts, such as (in, of), (<s>, of), (in, acm). "proc" is also the prefix of "proceedings", so we become increasingly confident that "proc" is a possible abbreviation of "proceedings".

5.3 Mining TemplateCollocation-SubCollocation Pairs (MTS):

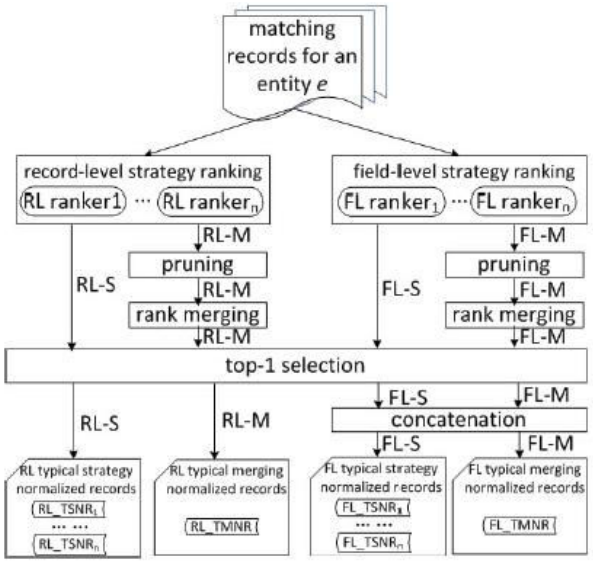
This module aim to find all template collocations and their subcollocations. The template collocations become the candidates with which it can expand (replace) the subcollocations. They will be used to generate the normalized component values for a field. Let an n-collocation tc be a template collocation and a k collocation kc be its subcollocation (k < n).

5.4 Mining Most Frequently Co-occurring Template Collocation:

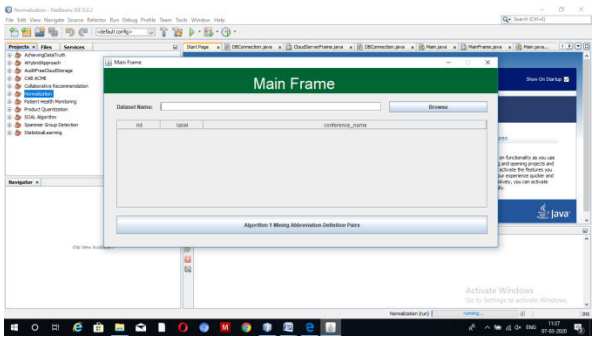
The above module, discussed how to obtain the template collocations and their corresponding subcollocations. We notice that some of the template collocations co-occur frequently. For example, among the values of the field venue, the template collocation "conference on" co-occurs most frequently with "in proceedings of the." We also observe that template collocation co-occurrence is not always bidirectional. For example, the template collocation "symposium on" co-occurs most often with "in proceedings of the", but "in proceedings

of the" co-occurs most frequently with "conference on."

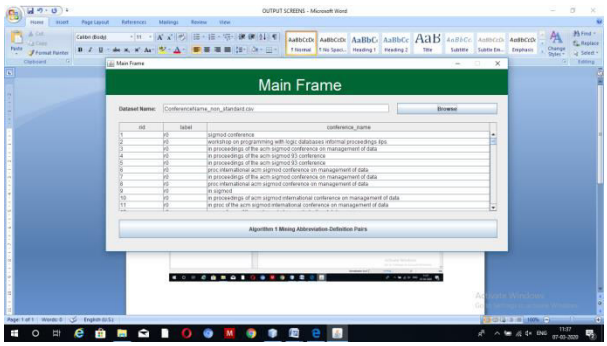
6.Architecture



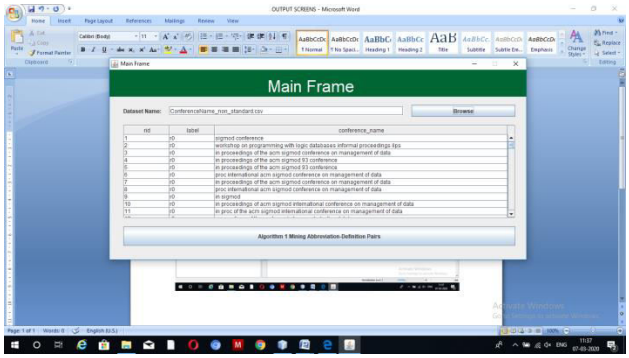
7. OUTPUT RESULTS



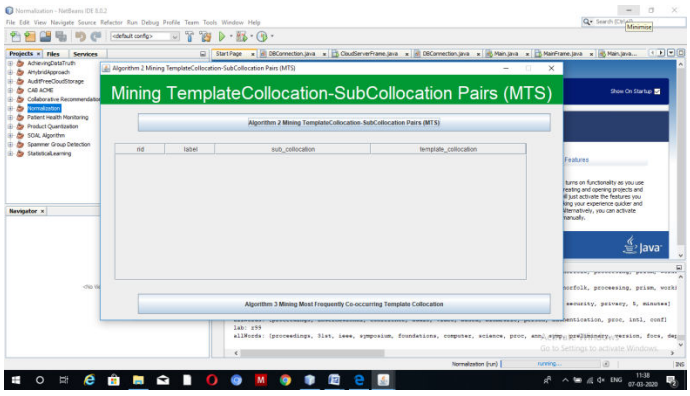
Load Dataset Screen



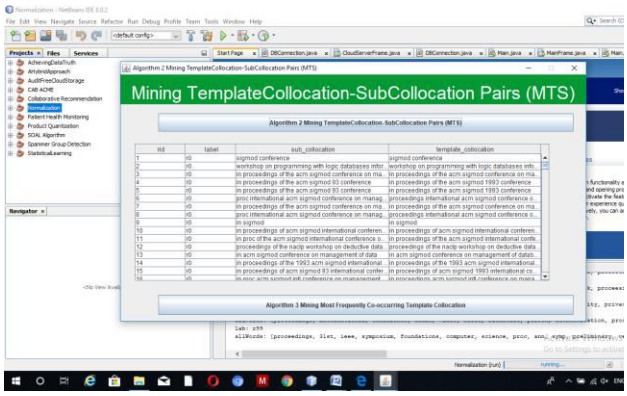
Show Dataset screen



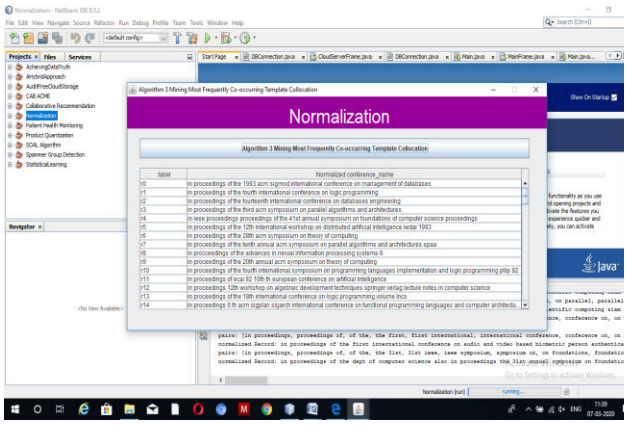
Implement Algorithm-1



Implement Algorithm-2



Minig Template collection-sub collection pairs screen



Implement Algorithm-3

8. CONCLUSION AND FUTURE ENHANCEMENT

This work studied the problem of record normalization over a set of matching records that refer to the same real-world entity. This work presented three levels of normalization granularities (record-level, field-level and value component level) and two forms of normalization (typical normalization and complete normalization). For each form of normalization, this work proposed a computational framework that includes both single-strategy and multi-strategy approaches. This work proposed four single-strategy approaches: frequency, length, centroid, and feature-based to select the normalized record or the normalized field value. For multistrategy approach, this work used result merging models inspired from metasearching to combine the results from a number of single strategies. This work analyzed the record and field level normalization in the typical normalization. In the complete normalization, this work focused on field values and proposed algorithms for acronym expansion and value component mining to produce much improved normalized field values. This work implemented a prototype and tested it on a real-world dataset. The experimental results demonstrate the feasibility and effectiveness of this approach. This method outperforms the state-of-the-art by a significant margin**.**

9. BIBLIOGRAPHY

1. Improving product marketing by predicting early reviewers on E-Commerce websites
S. Kodati, M. Dhasaratham, V. V. S. S. Srikanth, and K. M. Reddy, “Improving product marketing by predicting early reviewers on E-Commerce websites,” Deleted Journal, no. 43, pp. 17–25, Apr. 2024, doi: 10.55529/ijrise.43.17.25.

2. Kodati, Dr Sarangam, et al. "Classification of SARS Cov-2 and Non-SARS Cov-2 Pneumonia Using CNN." Journal of Prevention, Diagnosis and Management of Human Diseases (JPDMHD) 2799-1202, vol. 3, no. 06, 23 Nov. 2023, pp. 32–40, journal.hmjournals.com/index.php/JPDMHD/article/view/3406/2798, https://doi.org/10.55529/jpdmhd.36.32.40. Accessed 2 May 2024.

3. V. Srikanth, "CHRONIC KIDNEY DISEASE PREDICTION USING MACHINE LEARNING ALGORITHMS," IJTE, pp. 106–109, Jan. 2023, [Online]. Available: http://ijte.uk/archive/2023/CHRONIC-KIDNEY-DISEASE-PREDICTION-USING-MACHINE-LEARNING-ALGORITHMS.pdf

4. V. SRIKANTH, "DETECTION OF PLAGIARISM USING ARTIFICIAL NEURAL NETWORKS," International Journal of Technology and Engineering, vol. XV, no. I, pp. 201–204, Feb. 2023, [Online]. Available: http://ijte.uk/archive/2023/DETECTION-OF-PLAGIARISM-USING-ARTIFICIAL-NEURAL-NETWORKS.pdf

5. V. SRIKANTH, "A REVIEW ON MODELING AND PREDICTING OF CYBER HACKING BREACHES," IJTE, vol. XV, no. I, pp. 300–302, Mar. 2023, [Online]. Available: http://ijte.uk/archive/2023/A-REVIEW-ON-MODELING-AND-PREDICTING-OF-CYBER-HACKING-BREACHES.pdf

6. S. Kodati, M. Dhasaratham, V. V. S. S. Srikanth, and K. M. Reddy, "Detection of fake currency using machine learning models," Deleted Journal, no. 41, pp. 31–38, Dec. 2023, doi: 10.55529/ijrise.41.31.38.

7. "Cyberspace and the Law: Cyber Security." IOK STORE, iokstore.inkofknowledge.com/product-page/cyberspace-and-the-law. Accessed 2 May 2024.

8. "Data Structures Laboratory Manual." IOK STORE, www.iokstore.inkofknowledge.com/product-page/data-structures-laboratory-manual. Accessed 2 May 2024.

9. Data Analytics Using R Programming Lab." IOK STORE, www.iokstore.inkofknowledge.com/product-page/data-analytics-using-r-programming-lab. Accessed 2 May 2024.

10. V. Srikanth, Dr. I. Reddy, and Department of Information Technology, Sreenidhi Institute of Science and Technology, Hyderabad, 501301, India, "WIRELESS SECURITY PROTOCOLS (WEP,WPA,WPA2 & WPA3)," journal-article, 2019. [Online]. Available: https://www.jetir.org/papers/JETIRDA06001.pdf

10. V. SRIKANTH, "Secured ranked keyword search over encrypted data on cloud," IJIEMR Transactions, vol. 07, no. 02, pp. 111–119, Feb. 2018, [Online]. Available: https://www.ijiemr.org/public/uploads/paper/1121_approvedpaper.pdf

11. V. SRIKANTH, "A NOVEL METHOD FOR BUG DETECTION TECHNIQUES USING INSTANCE SELECTION AND FEATURE SELECTION," IJIEMR Transactions, vol. 06, no. 12, pp. 337–344, Dec. 2017, [Online]. Available:

https://www.ijiemr.org/public/uploads/paper/976_approvedpaper.pdf

12 . SRIKANTH MCA, MTECH, MBA, "ANALYZING THE TWEETS AND DETECT TRAFFIC FROM TWITTER ANALYSIS," Feb. 2017. [Online]. Available: http://ijmtarc.in/Papers/Current%20Papers/IJMTARC-170309.pdf

14 Srikanth, V. 2018. "Secret Sharing Algorithm Implementation on Single to Multi Cloud." International Journal of Research 5 (01): 1036–41. https://journals.pen2print.org/index.php/ijr/article/view/11641/11021.

5. K. Meenendranath Reddy, et al. Design and Implementation of Robotic Arm for Pick and Place by Using Bluetooth Technology. No. 34, 16 June 2023, pp. 16–21, https://doi.org/10.55529/jeet.34.16.21. Accessed 20 Aug. 2023.

16. Babu, Dr P. Sankar, et al. "Intelligents Traffic Light Controller for Ambulance." Journal of Image Processing and Intelligent Remote Sensing(JIPIRS) ISSN 2815-0953, vol. 3, no. 04, 19 July 2023, pp. 19–26, journal.hmjournals.com/index.php/JIPIRS/article/view/2425/2316, https://doi.org/10.55529/jipirs.34.19.26. Accessed 24 Aug. 2023.

17. S. Maddilety, et al. "Grid Synchronization Failure Detection on Sensing the Frequency and Voltage beyond the Ranges." Journal of Energy Engineering and Thermodynamics, no. 35, 4 Aug. 2023, pp. 1–7, https://doi.org/10.55529/jeet.35.1.7. Accessed 2 May 2024.

18. K. Meenendranath Reddy, et al. Design and Implementation of Robotic Arm for Pick and Place by Using Bluetooth Technology. No. 34, 16 June 2023, pp. 16–21, https://doi.org/10.55529/jeet.34.16.21. Accessed 20 Aug