

Citation of Court Cases with Shift-Or Pattern Matching

Omisore O. M. *, Samuel O. W

Department of Computer Science, Federal University of Technology, Akure, Nigeria

*Corresponding author: ootsorewilly@gmail.com

Received September 06, 2013; Revised October 08, 2013; Accepted October 10, 2013

Abstract Parties often appear in law courts with cases that have festered for long periods of time at great expense. This makes judges to face the challenge of overcoming impasses that frequently occur during adjudication. In developing countries, reports on previous judgments are kept piece meal in file cabinets and thereby giving edge to manual citation of court cases. This citation form had posed a great deal of problem such as, lengthened period between the time a case is opened and when accurate judgment is delivered, in the administration of Law and Justice. The advent of digital computers has made the routine use of pattern-matching possible in various applications and has also stimulated the development of many algorithms. In this study, a model that adopts Shift-Or Pattern Matching technique is proposed for citing authorities during legal jurisdiction. The model is implemented on Windows Vista Home Premium and WAMP APACHE as the web server. Record of court cases decided upon in the Ondo State High Court (OSHC), Akure between 2000 and 2012 was used to evaluate the performance of the system.

Keywords: court cases, case citation, crimes, information retrieval, Approximate String Matching

Cite This Article: Omisore O. M., and Samuel O. W, "Citation of Court Cases with Shift-Or Pattern Matching." *American Journal of Information Systems* 1, no. 1 (2013): 1-8. doi: 10.12691/ajis-1-1-1.

1. Introduction

In most countries, crimes are defined by statutory rules and regulations enacted by the law and are resolved by judiciary gathering in a courtroom [1]. Different types of punishment including: fines, imprisonment, disqualification from offices, and death; are ascribed with diverse crimes.

Courts are the central means of dispute resolution with a general notion "*all individuals have the right to bring their claims before a court*" [2]. Judicial settlement conferences present novel challenges to judges in assisting parties to settle their dispute rather than continue to litigate. Parties often appear in law courts with cases that have festered for long periods of time at great expense [3]. In many instances, the judge faces a particular challenge of overcoming impasses that frequently occur during the negotiations [4]. A successful settlement judge must employ creative approaches to bring about a resolution, particularly, when the negotiations appear at a dead end. The judge should be able to help the parties break through impasses with a process suggestion, additional information, or a settlement recommendation. Judges should therefore have a number of useful impasse-breaking techniques at their disposal [3].

Court facilities range from simple facilities in rural communities to sophisticated technology in cities, recent changes in courtroom technology of everyday life is affecting case dispositions increasingly quickly [5].

Anecdotal evidence suggests that electronically presented trials save from one-fourth to one-third of the time normally taken to try a similar case in a traditional

fashion. Most of our appellate systems require verbatim records when serious cases are appealed and judges have interest in the court records such that the records are accessible in a timely and accurate fashion. This aids them in clarifying factual and legal matters during trial processes [5]. However, conscientious and competent judges are best supported by accurate trial records [6]. Hence, the more accurate the record, the less likely a case will be reversed.

In legal reasoning, judges follow rules defined by written law but also include legal precedents in their decision process. Often, the interpretation of law varies to a large extent among judges and it is difficult to find a common ground [7]. A court decision is usually published in one or more reporters. These are series of published bounded volumes which are publicly or privately available to legal practitioners [8]. Legal profession is not immune to the wind of change as legal research is very dynamic. In recent times, there had been some pragmatic and progressive developments in legal research however citation of court cases in a fast, accurate and accessible manner is still lagging.

The use of Expert Systems (ESs) can be espoused to leverage the existing limitations in citation of court cases. Expert System (ES) is an intelligent interactive computer based decision tool that uses facts and rules to solve difficult real life problems. The development of ES is based on knowledge acquired from human experts in a particular field, it features user friendly interfaces which make them highly interactive in nature, and provide accurate and timely solutions to difficult real life problems [9,10,11].

Over the years, pattern-matching has been routinely used in various computer applications such as: text editors,

information retrieval, imaging analysis and searching nucleotide sequence patterns in genome and protein sequence databases [12]. This research therefore proposes a model that can cite court cases. The proposed model adopts Shift-Or Pattern Matching, an Approximate String Matching (ASM) technique that compares the characters of strings based on their distance. The model is driven by web based system for the purpose of citing court cases in an online and real-time manner.

The rest of this paper is organized such that: section 2 covers an overview of background study and related works; section 3 describes the proposed model and explains how the matching techniques can be adopted in the citation of court cases; section 4 presents an experimental study and evaluation of the model using the records obtained from OSHC, Akure, Nigeria. Lastly, section 5 presents the conclusion and future works.

2. Background Study and Related Works

This section presents an overview related studies on the application of Information Technology (IT) in citation of court cases.

Delay in legal proceedings due to manual searching of law reports causes longer time before cases' judgments are delivered [13]. Lawyers and judges often spend longer time to get matching authorities during litigation processes so as to backup their points. With computerization of law reports, a single storage media such as hard disk can house thousands of law reports [14]. In order to provide a means of reducing the volume of books that legal practitioners have to convey from their chamber to the law court at any point in time, a computer based law information system for storing and retrieval of legal materials was developed in [13]. The system provides a mechanism for quick references and citation of court cases by lawyers and judges. The application software did not provide means for direct searching of specific cases rather user will have to scroll through thousands of cases before he/she can access an case of interest.

During court sessions, litigants are represented by lawyer(s) who are versatile in the case(s) to be handled, and that of previous related cases [15]. Due to this, lawyers are expected to have anytime access to Law Reports for proper search. Direct search and search through law indexes remain the standard approaches to locate an authority in law reports [1,16]. In the former, access to law reports can be based on chronological and alphabetical indices. These techniques involve moving through different volumes of law reports and checking the index of authorities as contained at the back page of the publication [16]. Searching ends when a desired authority is located or such point cannot be defended. These steps, thereby, make the search methods cumbersome and time consuming [1]. As a result, search methods contribute to the lengthened time expended on legal matters.

Law Index is a publication that combined different volumes of Law Reports. This publication does not report authorities but points to the volume and pages of the reports for proper usages [16]. Since law index combined several volumes of law report, it makes searching for authorities more convenient. The complete index to Federation Weekly Law Report (FWLR) published

annually in Nigeria contains 52 volumes. A legal practitioner who wishes to cite an authority picks a volume of the complete index to an FWLR and search for the headings of the law he wishes to cite. This will direct him to the law report and pages of the report where such case is cited [14].

Law index had brought about a great relief to legal practitioner who would not have to pick several volumes of law report before getting an index to a desired authority; yet it has its limitation. A single volume of law index contains limited volumes of law report hence as years roll by, several volumes of law reports are available [15]. As law index grows with years passing, it is not wondrous having a situation that will be as bad as that of direct searching. Also, to locate an authority using law index, the legal practitioner searches through several legal headings/categories arranged in alphabetical order.

Information retrieval is the activity of retrieving stored data that provides information to support user's anomalous state of knowledge [17]. Professionals who are subject specialists in their task domain are distinguished from clerks based on document handling [18]. Lawyers are professionals who have refined their tools and work routines for many years [19]. They are also classified as experts in legal matters. Legal text retrieval has held a central position in information retrieval; it is concerned with legal norms which are subject to continuous interpretation and argumentation by lawyers. Reference [18] developed a prototype Information Retrieval System (IRS) with the aim of investigating how to provide lawyers with relevant recorded knowledge. The research demonstrated a facility for the use of terms in Thesaurus as keywords with visible references to documents of special interest using *Reversed Indexing*. Reversed indexing offers a direct way of retrieving relevant documents but it is time consuming [20].

Some of the legal practitioners in developing countries are intemperate with their scorn, one account notes that "many law librarians were appalled to learn that computer-assisted legal research would operate free of their dearly beloved, elaborate structure of indexes and digests" [21]. Technology came to courthouses long ago; today jurisdictions throughout Australia, Canada and the United State are using Case Management Systems and experimenting with Electronic Filing Systems [5]. High level of computer illiteracy has being an impacting factor that slows down the operation of cyber court as most parties, judges and court administrators are still trying to grasp the technologies on individual basis [22]. In the non-developed parts of the world, human attitudes toward invention of IT are very poor [14]. Reference [13] highlighted the need to develop a computer based law information system for storing and retrieving legal materials. This was to provide a mechanism for quick reference and citation of court cases during court sessions.

The influence of Computer-Assisted Legal Research (CALR) on legal issues and court citations had become an issue of debate among legal practitioners. Some law professionals are of the opinion that CARL is not effective as the conventional methods hence it should not be adopted in legal issues. In 2005, [9] clarified that CALR does not affect courts in the same way it is affecting other parts of the legal profession. Hence, commentators who asserted that CALR is reshaping the law are of weak

position. CALR does not change the results of legal research rather it improves litigation processes with better speed and accuracy [9].

Previously, the monitoring and citation of court cases have been carried out manually. This takes great amount of time and other resources which raised eyes brows towards the need for the development of an indigenous database package for the effective and efficient administration of justice in Nigeria [15]. A software package christened 'CaseLaw' was developed in a two-tier architecture using Microsoft Access Database Management System and Microsoft Visual Basic 6.0. The model composed of three major files capable of keeping data of lawyers, sets of cases handled by the lawyers, and the details of each case as found in law reports. The model was tested in a Local Area Network environment.

ESs are designed to solve complex problems by reasoning about knowledge in human's nature rather than following the procedure of a developer as in the case of conventional programming [11]. Neural Network (NN) and Fuzzy Logic (FL) appear to be the major techniques adopted for developing ESs [23]. Computer models can be developed in line of human thinking concepts such that they are used in recognition, categorization, and analogy making. The major part of legal reasoning is formally interpreted as analogy making process using methods that incorporate the ability to specify likelihood with known court decisions. As a result, modeling expert systems that can attend to real life issues in Law and Justice is necessary [9]. Model for analogy and decision making in citing court cases was proposed in [9]. The model adopted NN and FL as its brain builder.

The basic purpose of legal citation is to allow legal practitioner locate cited sources accurately and efficiently. Case citations follow a standard convention in widely accepted format. However, variations in citation styles of portfolio data and metadata extracted from the text of a decision shows that the program logic would need to be highly flexible. There is need to develop repositories for storing decision text and metadata before they are linked. FL has been used to resolve court citation issues by creating links to previous court decisions [9].

High technology courtrooms have been public initiatives that are aimed at assessing the value of technology so as to improve the overall efficiency of courts and advance the sophisticated administration of court issues. The emergence of these technologies has assisted in processing, reviewing, distributing and storing large amount of court data on Internet [24]. Jurisdiction can also be exercised via the World Wide Web as the case of Caitlin, in New York City [22].

String matching algorithms are an important class of algorithms that checks the occurrence of string subsets, called patterns, in larger string called text [25]. A variety of algorithms for searching patterns have been proposed. Reference [24] describes a pattern-matching algorithm which finds all occurrences of a pattern of length m within a text of length n in $O(m + n)$ units of time. The algorithm does not without back up the input text hence, it only requires $O(m)$ locations of internal memory if the text is read from an external file, only $O(\log m)$ units of time will elapse between consecutive single-character inputs. All of the constants of proportionality implied by these "O" formulas are independent of the alphabet size [25].

The advent of digital computers has made the routine use of pattern-matching possible in various applications [12]. This has also stimulated the development of many algorithms. Large number of algorithms is known to exist to solve string matching problem [25]. Based on the number of patterns searched for, the algorithms can be classified as single pattern and multiple pattern algorithms. The algorithms scan text with the aid of a window whose size is equal to the pattern's length. The present day pattern-matching algorithms match the pattern exactly or approximately within the text [12]. An exact pattern-matching is to find all the occurrences of a pattern ($x = x_1, x_2, \dots, x_m$) of m -characters in a text ($y = y_1, y_2, \dots, y_m$) of n -characters which are built over a finite set of characters of an alphabet set denoted by Σ and the size of this set is equal to σ . This approach is commonly known as a brute-force method.

Approximate string matching consists of finding all approximate occurrences of pattern x in text y with of length m and n respectively [27]. Approximate occurrences of x are segments of y that are close to x according to a specific distance which must be not greater than a given integer k . Hamming and Levenshtein distance are commonly considered [28]. The former is known as approximate string matching with k mismatches while the later is known as approximate string matching with k differences.

The understanding regarding citation of legal authorities during jurisdiction has been identified as information processing [18]. Numerous studies have been carried out in the field of information processing yet it has not being properly applied for citing court cases. Hence, in this study experiments how shift-or pattern matching can be adopted for citation purpose in courtrooms and other public or private law chambers.

3. Citation of Court Case Using Shift-Or Pattern Matching

This section presents the architecture of Citation of Court Cases using shift-or pattern matching.

3.1. Court Case Citation Architecture

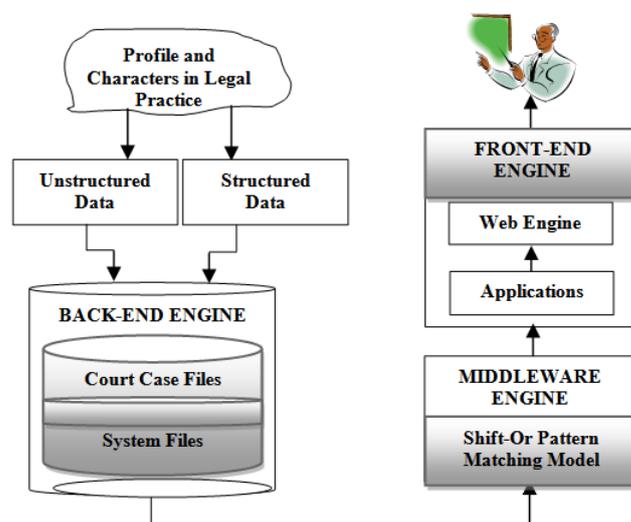


Figure 1. Architecture for Citing Court Cases using Shift-Or Pattern Matching

The architecture contains of three main components as shown in Figure 1. These are the: front-end engine, the back-end engine, and middleware engine.

Front-end engine is composed of the application modules and web engine with which the users of the system can interact with it. The modules coordinate users' access to the data stored in the system's database while the web engine aids creation and management of system data in an online and real-time mode. The data required by the citation architecture are captured and encoded in text formats which are kept in the back-end engine. Example of captured data includes details of cases that had been decided in some law courts, and the profile and characters of agents involved in legal practice.

The shift-or matching does the actual processes for a chosen transaction in order to cite relevant case and this is performed by the middleware engine.

3.2. Shift-Or Pattern Matching Algorithm

The Shift-Or pattern matching algorithm which was proposed by [29] is widely employed in text comparisons, database search, and pattern matching. The algorithm is very fast in practice and very easy to implement [27]. It overhauls both *Hamming* and *Levenshtein* distance of ASM. *Shift-Or* Pattern-Matching is thereby adopted by this research for the purpose of citing relevant authorities during legal jurisdictions. The explanation of the algorithm given in this paper is exact string-matching problem, however, a detailed explanation on how it can handle the cases of k mismatches and of k differences (insertions, deletions, or substitutions) is given in [27].

Given a pattern \mathbf{x} and text stream \mathbf{y} built by concatenating m and n characters of the finite alphabets Σ with size σ . All distinctive characters in Σ are generated as:

$$\Sigma = \bigcup_x^y \quad (1)$$

For each character of Σ , a state bit S_c is derived as:

$$S_c[i] = \begin{cases} 0 & \text{if } x[i] = c \\ 1 & \text{otherwise} \end{cases} \quad (2)$$

Where $S_c[i]$ is the bit array for character c of the Σ as related to its position in \mathbf{x} . Each bit array has a size which is the same size as that of the pattern to be searched for. After building the state bits, a matching table \mathbf{T} with dimension $m \times n$ is that holds information of all matches between the characters of \mathbf{x} and \mathbf{y} is generated. This is done by comparing the characters $\mathbf{x}[i]$ and $\mathbf{y}[i]$ using the bitwise-logical operators known as Shift-Or. The operation is done as:

$$\mathbf{T}_{j+1} = \mathbf{T}_j \gg 1 \text{ OR } \mathbf{S}_y[j+1] \quad (3)$$

Where $\mathbf{i} = (0, 1, \dots, m)$, and $\mathbf{j} = (0, 1, \dots, n)$. At $\mathbf{j} = 0$, an initial value \mathbf{T}_0 is set as the table value, this denotes an empty matching table. Then for each character $\mathbf{x}[i]$ in the pattern, starting from $\mathbf{x}[0]$, a comparison is made with character $\mathbf{y}[i]$ in the text, if a match is reported the initial value is shifted one bit downward and stored as \mathbf{T}_j , the next character $\mathbf{x}[i+1]$ in the pattern is taken against the next character $\mathbf{y}[i+1]$ in the text. If the comparison

process returns a no-match, the initial value \mathbf{T}_0 is stored as the new value \mathbf{T}_j , and another comparison process is observed between the characters $\mathbf{x}[0]$ and $\mathbf{y}[i+1]$.

A success check is performed each time the matching reports success that is when $\mathbf{x}[i] = \mathbf{y}[j+1]$, the bit at $\mathbf{T}_j[m-1]$ is checked. Once $\mathbf{T}_j[m-1] = 0$, the process is completed and an exact match of \mathbf{x} is reported to be found in \mathbf{y} . Otherwise the matching processes continue till the characters of string \mathbf{y} are exhausted. If a match is found, the matching start-position P_0 and end-position P_1 are calculated using:

$$P_0 = (j - m) + 1 \quad (4)$$

$$P_1 = (P_0 + m) - 1 \quad (5)$$

Where m is the length of \mathbf{x} . The whole matching process is diagrammatically conceptualized as Figure 2.

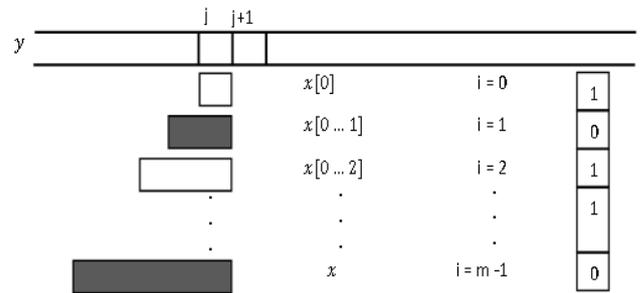


Figure 2. Conceptual Diagram of Matching Processes between Pattern \mathbf{x} and text stream \mathbf{y}

For instance, to search for a pattern $P = GATAA$ from text $T = CAGATAAGAGAA$

$$m = 5; n = 12; \Sigma = \{A, C, G, T\}$$

The state bits of each character in the pattern are generated using Eq (1) as shown in Table 1, while the matching information for searching the pattern \mathbf{P} in the text stream \mathbf{T} is shown in Table 2. It is clear that from Table 2, the bit at $\mathbf{T}_{j=6}[m-1] = 0$. Hence, at \mathbf{T}_6 [4], an occurrence of \mathbf{x} has been found. The values of P_0 and P_1 are computed using equation 4 and 5 respectively as:

$$P_0 \Rightarrow (j - m) + 1 \Rightarrow (6 - 5) + 1 = 2.$$

$$P_1 = (P_0 + m) - 1 \Rightarrow (2 + 5) - 1 = 6.$$

Therefore the pattern \mathbf{P} which occurs in text \mathbf{T} is a substring found between positions 2 to 6.

Table 1. State Bits Generated for character of Σ

S_A	S_C	S_G	S_T
1	1	0	1
0	1	1	1
1	1	1	0
0	1	1	1
0	1	1	1

3.3. Adopting Shift-Or in Court Case Citation

Court cases are cited by matching the search-key(s) found in a system generated query against appropriate relations of court case files. If a match is found, tuples of the case record(s) whose ID matches the search-key(s) are

then displayed for the user’s consumption. The search-key and court case files are taken to be respectively synonymous to the pattern and text stream explained above. Therefore, the matching process described in section 3.2 is established between the pattern and text stream whenever there is call for citation of court cases.

The Shift-Or matching technique is adopted in this study to perform multiple pattern check. Hence, a system query can be generated from one or more pattern(s). The patterns from which cases can be cited are shown diagrammatically in Figure 3.

Table 2. Matching Table for Searching Occurrence of P in T

	j	0	1	2	3	4	5	6	7	8	9	10	11
i	C	A	G	A	T	A	A	G	A	G	A	A	A
0	G	1	1	0	1	1	1	1	0	1	0	1	1
1	A	1	1	1	0	1	1	1	1	0	1	0	1
2	T	1	1	1	1	0	1	1	1	1	1	1	1
3	A	1	1	1	1	1	0	1	1	1	1	1	1
4	A	1	1	1	1	1	1	0	1	1	1	1	1

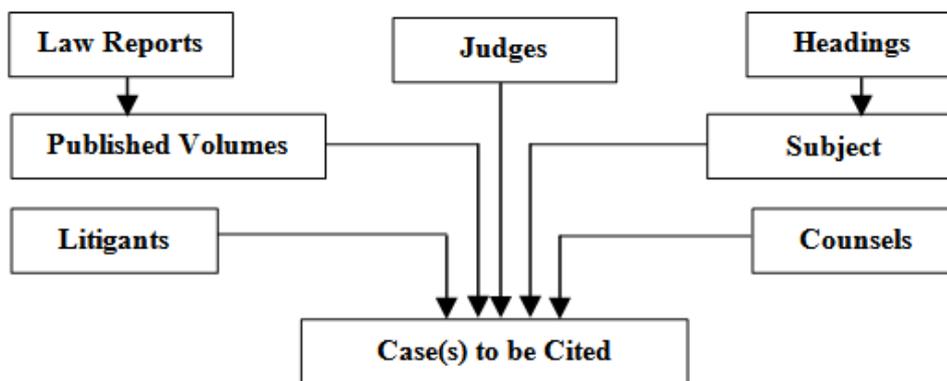


Figure 3. Pattern Location during Case Citation

4. Experimental Study and Evaluation

This section presents the features of the environment on which the study was carried out. The description of data which was used to evaluate the efficiency of the proposed model is as well given in this section.

4.1. Experimental Settings

The proposed model is implemented as a web based system with three-tier architecture. The web system consists of three main components which are the web browser agent, the web server application, and the database server which are well installed and configured independently.

The front-tier is the end through which the user communicates with the application. It requires the installation of a web browser at the client system. Fortunately, a web browser accompanies almost all operating systems in use today. Hence, the user is only required to launch the browser and specify the address of the web application: [http://127.0.0.1/Case Citation System/index.php](http://127.0.0.1/Case%20Citation%20System/index.php) in the address bar of the browser. The front-end is designed with HTML tags that give the structure of a page while Java Scripts determine the behaviour of the page.

HTML documents are simple to learn and developers, irrespective of the level, can code classy web systems with minimal efforts [30]. Also, there are several free editors that support HTML tags. With JavaScript, small snippets

of program code can help to fulfill complex tasks. Its interpreter is built into every scriptable browser [31]. Hence, there is no need of acquiring any additional tool so as to utilize the proposed system.

The middle-tier is the end through which users’ requests are processed. It requires the presence of web application server that can listen to connection requests from client computers and direct the requests to the corresponding web application module for service. The middle-tier also requires a network connection to the database server at the back end. The middle-tier is implemented using Hypertext Preprocessor formally known as Personal Home Page (PHP).

PHP has its Object Oriented features combing those of Java and C# languages. PHP runs 5 to 20 times faster than Java. It is extremely easy to use in developing complex web applications, in a considerable period of time. PHP is the real gem of all scripting languages with a huge number of users worldwide [32].

The back-end hosts the data used by the web application and it is managed by the System Administrator. In this research, the back-end is hosted using MySQL DBMS, an open source application. The System Administrator performs administrative tasks such as the creation and backing up of database, recovery in case of database failure, data and system tuning.

MySQL DBMS is a fast Relational DBMS with the functionalities of varying leading database applications; it does not carry a hefty price tag as it is an open source application which can be easily downloaded [33].

The design of the web based system was developed to run in either a Localhost characterized by WAMP Server

or in an Internet environment with a view to ensuring online and real time access to previously decided court cases. The Localhost environment characterized by Windows Apache MySQL PHP (WAMP) Server is adopted in this project, however other platforms like Linux has LAMP providing the same services in Linux based machines. WAMP is an open source application that allows users to run web based applications on their local machine just the same way the applications behave in an Internet environment.

4.2. Data Description

For the purpose of evaluation of the model proposed in this study, the list of cases administered at the OSHC,

Akure between January, 2000 and August, 2012 were collected and analyzed. Visitations were made to the Court for the purpose of interviewing some legal staff of the court. After a series of interview, only criminal cases decided in the period of study was extracted for this study. A total of 442 cases that were extracted are summarized in Table 3. The details of each case are entered in their raw description in text files and stored in the database component of the architecture shown in Figure 1. In each file, cases are cited by taken any of the characters in Figure 3 as pattern and the case files as text stream.

The representation of the case types are: (a) Forgery and Stealing; (b) Man Slaughter; (c) Armed Robbery; (d) Rape and Kidnap; (e) Conspiracy; and (f) Murder.

Figure 4. Web Form for Citing Court Case

Session Id: 01		Total Number of Cases Cited: 36			Number of Relevant Cases: 5		
First Page Previous Page 1 - 10 of 36 First Page Last Page							
S/No.	Suit No.	File No.	Subject Matter	Date	Case Litigants	Legal Adviser Position	Decision
1	CA/C/223/168	OD/102/90	Murder	2001-01-20	Olumilude Christopher Vs. The State)	Whether the conviction on accused for the offence is confirmed	Initiate a settlement proceedings out of court
2	CA/OD/258/2001	OD/006/01	Murder and Conspiracy	2002-03-21	Adesina Kayode Vs. The State	The evidence of prosecution was cogent enough and met the standard of proof beyond reasonable doubt to sustain teh conviction for murder	No Scintilla of provocative attack found/proved by the deceased. The accused is therefore convicted and sentenced to death by hanging
3	CA/OD/24M/90	OD/001/2002	Murder and Conspiracy	2003-03-04	Stephen Okafor Vs. The State	The death of the deceased was caused by the fist blow inflicted on him by the accused person	The death of the deceased was not caused by the fist blow as shown by medical evidence, but by laceration of the liver. Accused accordingly discharged and acquitted
4	SC.12/200	OD/002/04	Murder and Conspiracy	2004-01-12	Adesina Olumide Vs. Aderinde Oluwafemi	Litigate the case to a logical conclusion	Accused person not guilty. Acussed discharged and acquitted. Sum of N450,000:00 be paid to the Accused
5	OD./28/99	OD/007/03	Murder and Conspiracy	2004-08-10	Alguokhliou Chidimmeka Vs. The State	Litigate the case to conclusion	Accused person guilty of murder. Acused to be executed by hanging
6	CA/I/97/2005	OD/021/10	Murder and	2006-05-08	Akeem Afolahan	That the peculiar circumstance is	The accused is convicted and

Figure 5. Details of Cases Cited for Session-05

Table 3. Criminal Cases reported at Ondo State High Court, Akure between January, 2000 and August, 2012

Type	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	Total	%
a	2	1	3	0	4	3	5	6	3	1	5	2	3	41	9.10
b	0	2	0	0	6	14	1	2	1	0	5	0	4	36	8.00
c	0	0	5	0	26	34	25	15	4	0	36	21	24	192	42.67
d	0	0	0	0	3	3	0	2	0	0	0	1	3	12	2.67
e	2	0	1	0	3	0	2	7	3	1	16	7	44	87	19.33
f	4	0	0	0	10	15	7	6	1	0	12	6	20	81	18.22
Total	8	3	9	0	52	69	40	38	12	2	74	37	98	442	100

4.3. Experimental Results and Evaluation

Cases can be cited with the aid of the web form displayed in Figure 4. The form is designed to cite cases using both alphabetic and chronological approaches; hence it demonstrates a degree of flexibility for users.

To evaluate the performance of the proposed system, the legal practitioners that works at both chambers of O. J. Jejelola and Michael Owolabi were given access the web based system for citing cases during their jurisdictions. This was observed between May to August, 2013.

A sample report generated by the system when used during a session is displayed as Figure 5. The report gives the number of cases that are related to the users' quest. For the purpose of evaluation, the user is made to go through the cases cited by the proposed system so as to confirm the cases that are both relevantly and irrelevantly cited. This is used to determine the Degree of Confidence (DoC) of the citations made by the model. The responses of the legal practitioners after 10 sessions with the proposed system are, as presented in Table 4, extracted and used to evaluate the performance of the proposed system.

In information retrieval context, standard measures, like precision, recall and receiver operating characteristic have been used to determine the performance of several systems [34]. In this study, DoC is taken as a measure to establish the confidentiality of the system. This is done by performing relational join operation on the result of the proposed system and the comments of legal practitioners. The DoC of a citation is determined as follows:

$$DoC_i = \frac{\text{Relevant Cases Cited}}{\text{Total Cases Cited}} * 100 \quad (6)$$

Where relevant cases are is number of cases cited by the model and one/more legal practitioner(s) confirm to be truly related to a case upon which the system is used, the total cases cited by the system can include irrelevant cases that is, the cases cited by the system but none of the legal practitioner confirms the relevancy of the case. The DoC of citations made during session 01 is computed as follows:

$$DoC_{S-01} = \frac{36}{41} * 100 = 87.8\%$$

The same procedure was observed to compute the DoC of citations made in other sessions too. To determine the Mean Accuracy (MA) and the Efficiency of the system, the error in citations made in each session was derived as:

$$Error_i = \frac{100 - DoC_i}{100} \quad (7)$$

Table 4. Response from Legal Practitioners

Session Id	Relevant Cases Cited by System	Irrelevant Cases Cited by System	DoC (%)	Error _i	1-Error _i
01	36	5	87.8	0.122	0.878
02	14	0	100.0	0.000	1.000
03	54	6	90.0	0.100	0.900
04	21	1	95.5	0.045	0.955
05	39	4	90.7	0.093	0.907
06	18	2	90.0	0.100	0.900
07	44	4	91.7	0.083	0.917
08	56	5	91.8	0.082	0.918
09	27	3	90.0	0.100	0.900
10	30	3	90.9	0.091	0.909

Therefore, the MA is calculated as:

$$MA = \frac{\sum_{i=1}^n (1 - Error_i)}{n} \quad (8)$$

$$MA = \frac{9.184}{10} = 0.9184$$

$$Efficiency = MA * 100 \quad (9)$$

$$Efficiency = 0.9184 * 100 \Rightarrow 91.84\%$$

Therefore, it can be inferred from the statistical analysis shows that the proposed system is 91.84% efficient in providing relevant citation using the Shift-Or Pattern matching technique.

5. Conclusions

In this paper, we have proposed a system that can aid legal practitioners such as judges and lawyers during their jurisdiction. In particular, a comprehensive model for citing relevant cases during court sessions is focused. This model is not meant to replace the orthodox method rather as an augmentation to it. The objective of the experimental study is to demonstrate the functional capability of the system with a view to gaining confidence of legal practitioners and the administrators of Law and Justice who hand round as the end-users of the system.

The paper also described Shift-Or, a pattern matching technique, as an active platform that can be adopted for processing court data. We also describe web services as a suitable medium used for intelligent Information Retrieval, the citation of court cases in an online and real-time approach and, as well as for consistent presentation and

management of data. The experiments conducted showed that the web based court citation system as here proposed has been proved to be both workable and effective with an acceptable level of accuracy in citing relevant cases hence, existing courtrooms in the developing part of the world should be encouraged to adopt this system during their court sessions.

It is clearly shown that the deployment of IT facilities in legal practices had drastically reduced the stress passed through while adjudging a matter and had brought a better result therefore, soft computing tools such as Genetic Algorithm, Fuzzy Logic, and Neural Networks can be applied to court cases citation.

Acknowledgement

We wish to acknowledge the staff of Ondo State High Court, Akure, Ondo State, Nigeria, for their assistance in the area of data collection and analysis, and the chambers of O. J. Jejelola, and Michael Owolabi for their contributions.

References

- [1] Omisore M. Olatunji and Samuel O. Williams, *Design of Internet Model for Court Case Citation*, Journal of Communication and Computer, "In Press".
- [2] Walker David, *The Oxford Companion to Law*, Oxford University Press, Pg. 301, 1980.
- [3] Morton Denlow *Breaking Impasses in Judicial Settlement Conferences: Seven (More) Techniques for Resolution*, Court Review - The Journal of the American Judges Association, Vol 46, Issue 4, Pg 130-4, 2009.
- [4] Morton Denlow, *Breaking Impasses In Settlement Conferences: Five Techniques for Resolution*, JUDGES' J., Fall 2000.
- [5] Fredric I. Lederer, *Courtroom Technology from the Judge's Perspective*, Journal of the American Judges Association, American Judges Association Vol. 35, No. 1, Pg. 20-24, 1998.
- [6] James A. Maher, *National Center for State Courts, Do Video Transcripts Affect the Scope of Appellate Review? An Evaluation in the Kentucky Court of Appeals*, 1990.
- [7] Jurgen Hollatz, *Analogy Making in Legal Reasoning with Neural Networks and Fuzzy Logic*, Artificial Intelligence and Law, Kluwer Academic Publishers, Netherlands, Vol. 7, Pg. 289-301, 1999.
- [8] Jack Rugh and Julia Lennen, *Using Fuzzy Logic to Create Links: Resolving References to Court Decisions*, Idea Alliance, Senior Publishing Specialist, Media and Publications Division, pg 1-14, 2003.
- [9] Hellyer Paul, *Assessing the Influence of Computer-Assisted Legal Research*, Law Library Journal, San Jose University, Vol. 97, No 2, Pg. 285-298, 2005.
- [10] Savory, S. E., *Artificial Intelligence and Expert Systems*, Ellis Horwood Series in Artificial Intelligence, Halsted Press, a division of Wiley, Chichester, England, pp. 28-29, August, 2006.
- [11] Samuel, O. W., Omisore, M. O., and Ojokoh, B. A., *A Web Based Decision Support System Driven By Fuzzy Logic for the Diagnosis of Typhoid Fever*, Expert Systems with Applications 40 (2013) 4164-4171.
- [12] Sheik, S., Aggarwal S., Poddar A., Balakrishnan N., and Sekar K., *A FAST Pattern Matching Algorithm*, J. Chem. Inf. Comput. Sci., Vol. 44, pp. 1251-1256, 2004.
- [13] Oladipupo B., *Law Information System: Searching and Retrieval of Legal Materials*, Computer Based Law Database on Compact Disc, Private Communications, 2001.
- [14] Irene Baghoomians and Frederick Schauer (2009), *Thinking Like a Lawyer: A New Introduction to Legal Reasoning*, Harvard University Press, Cambridge (Ma), pp239.
- [15] Akinyokun O. Charles, *Computer Aided System for Monitoring and Citation of Court Cases*, Law and Practice Journal; Nigerian Law School, Abuja, Nigeria, 2006.
- [16] Adegoke M. A. (2004), *Development and Implementation of a Computer-Aided System for Monitoring and Citation of Court Cases*, Master Thesis, Federal University of Technology, Akure.
- [17] Belkin N. J., Oddy R. N., and Brooks H. M. (1982), *ASK for Information Retrieval: Background and Theory*, Journal of Documentation, vol. 3 issue 8, Pg. 61-71.
- [18] Morten Hertzum, Henrik Soes, and Erik Frokaer, *Information Retrieval Systems for Professionals: A Case Study of Computer Supported Legal Research*, European Journal of Information Systems, vol. 2, No. 4, pp. 296-303, 1993.
- [19] Tenopir C, Full-Text Databases, *Annual Review of Information Science and Technology* vol. 19, 215-246, 1984.
- [20] Foskett D J, *Thesaurus*. In *Subject and Information Analysis* (DYM E D, Ed.), pp. 270-316. Marcel Dekker, New York, 1985.
- [21] Harrington, *supra* note 1, at 546.
- [22] Lucille M. Ponle, *Michigan Cyber Court - Bold Experiment in the Development of the First Public Virtual Courthouse*, North Carolina Journal of Law and Technology, University of Michigan, Vol. 4, Issue:1, Pg: 51-60, 2002.
- [23] Samuel, O. W. and Omisore M. O., *Hybrid Intelligent System for the Diagnosis of Typhoid Fever*, Journal of Computer Engineering and Information Technology, Journal of Computer Engineering and Information Technology, vol. 2, issue 2, 2013.
- [24] Gwenn M. K., *From the Internet to Court: Exercising Jurisdiction over World Wide Web Communications*, Fordham Law Review, Vol. 65, No. 5, Article 8, Pg. 4-25, 1997.
- [25] Vidya SaiKrishna1, Akhtar Rasool, and Nilay Khare, *String Matching and its Applications in Diversified Fields*, International Journal of Computer Science Issues, Vol. 9, Issue 1, No 1, 1694-0814, 2012.
- [26] Knuth Donald, Morris James, and Pratt Vaughan, *Fast Pattern Matching in Strings*, SIAM J. COMPUT., Vol. 6, No. 2, 1977.
- [27] Maxime Crochemore and Thierry Lecroq, *Pattern Matching and Text Compression Algorithms*, Technical Report, Institut Gaspard Monge, Universite' de Marne la Valle'e, 2 rue de la Butte Verte, F-93166 Noisy-le-Grand, Cedex, France.
- [28] Wu, S. and Manber, U., *Fast Text Searching Allowing Errors*, Communication of ACM, vol 35, Issue 10, pp. 83-91, 1992.
- [29] Baeza-Yates, R. A., Gonnet, G. H., *A New Approach to Text Searching*. Communication of ACM vol. 35, no. 10, pp. 74-82, 1992.
- [30] Robert G. Fuller and Laurie Ann Ulrich (2004), *HTML in 10 Simple Steps or Less*, Wiley Publishing, Inc, Indianapolis.
- [31] Danny Goodman, *JavaScript Bible*, 4th Edition, Hungry Minds Publishing Inc, Indianapolis (United States), 2001, pp. 3-4.
- [32] Greenspan Jay and Brad Bulger (2001), *MySQL-PHP Database Applications*, M&T Books Worldwide, Inc. New York, pp. 71-94
- [33] Maslakowski Mark and Butcher Tony (2000), *Teach Yourself MySQL in 21 Days*, Sams Publishing, New Jersey, United States.
- [34] Herlocker, J. L., J. A. Konstan, A. Borchers, and J. Riedl, *An Algorithmic Framework for Performing Collaborative Filtering*, In Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99).