FEDERAL UNIVERSITY OF AGRICULTURE
ABEOKUTA NIGERIA

68TH INAUGURAL LECTURE

BIGGER AND BETTER DATABASE FOR INNOVATIVE ADVANCEMENT IN INDUSTRIAL REVOLUTION

by

Professor Adio Taofiki Akinwale

M.Sc (Wrocław), Ph.D (Wrocław), M.Sc (Warsaw), Ph.D (Łódź), Poland

(Professor of Computer Science)

Department of Computer Science

College of Physical Sciences

BIGGER AND BETTER DATABASE FOR INNOVATIVE ADVANCEMENT IN INDUSTRIAL REVOLUTION

by

Professor Adio Taofiki Akinwale

M.Sc (Wrocław), Ph.D (Wrocław), M.Sc (Warsaw), Ph.D (Łódź), Poland

(Professor of Computer Science)

Department of Computer Science

College of Physical Sciences

FUNAAB INAUGURAL LECTURE
Series No. 68

Wednesday, November 24, 2021

FUNAAB INAUGURAL LECTURE

Series No. 68

by

Professor Adio Taofiki Akinwale
(Professor of Computer Science)

The 68th Inaugural Lecture was delivered under the Chairmanship

of

The Vice-Chancellor

Professor Felix Kolawole Salako
B. Agric (Soil Science), M.Sc (Soil Physics/Soil Conservation)(Nigeria), Ph.D (Ibadan), FSSN, FASN

Published November, 2021

ISBN:978-978-991-965-9

FUNAAB INAUGURAL LECTURE SERIES

Professor Adio Taofiki Akinwale
M.Sc (Wrocław), Ph.D (Wrocław), M.Sc (Warsaw), Ph.D (Łódź), Poland
(Professor of Computer Science)
Department of Computer Science
College of Physical Sciences
Federal University of Agriculture, Abeokuta, Nigeria

BIGGER AND BETTER DATABASE FOR INNOVATIVE ADVANCEMENT IN INDUSTRIAL REVOLUTION

The Vice Chancellor
Past Vice Chancellors
Deputy Vice Chancellor(Academic)
Deputy Vice Chancellor (Development)
Past Deputy Vice Chancellors
Members of the Governing Council
The Registrar
The Bursar
The University Librarian
Deputy Dean, College of Physical Sciences
Deans of other Colleges and Dean Post Graduate School
Directors of Units, Institutes and Centres
Head, Department of Computer Science
Heads of other Departments
Distinguished Members of the University Senate
All Academic Staff
Visiting Academics and Colleagues
Esteemed Administrative Staff
Members of my Family
Captains of Industries
Cherished Friends and Guests of the University
Distinguished Ladies and Gentlemen
Gentlemen of the Press and Great FUNAABITES

1.0 Introduction

Vice Chancellor Sir, it is a great pleasure and honour to stand before you and your Management Team today to give my inaugural lecture as a Professor of Computer Science in the Department of Computer Science, College of Physical Sciences of this great and unique Federal University of Agriculture, Abeokuta. My journey as an Academic in this great University started as Lecturer I on October 28, 1998 and culminated in my appointment as Professor in 2012. This is the first inaugural lecture from the Department of Computer Science and the second from the defunct Mathematical Sciences Department from which Department of Computer Science emerged in 2005/2006 session. The first titled "Total quality: a mathematical panacea for national productivity improvement" was delivered by Prof Adewale Roland Tunde Solarin in 1999. This is also the third inaugural lecture from the College of Physical Sciences, coming after those of Prof Joseph Adeniyi Olowofela and Prof (Mrs) Catherine Oluyemisi Eromosele of Department of Physics and Department of Chemistry respectively.

Inaugural Lecture is an opportunity for newly promoted or appointed Professors to inform Colleagues in the University and the general public about their research career so far, and update Colleagues on their current and future research directions. I am very grateful for the opportunity to present the 68th inaugural lecture on what I may call a harvest of my research contributions summarily framed under the title "Bigger and better database for innovative advancement in industrial revolution". The topic was chosen to enable me throw some light on my field of specialization in computer science (knowledge database system) and my contributions to learning, research and service to the University and the wider community. With every stretch of humility, I crave your indulgence to join me as we explore my research work and findings over the years.

1.1 Formation of database

The term data in database is the fact or figure that can be analysed or used in an effort to gain knowledge or make decision. Data is the plural of datum, which is rarely used, and it means 'what is given' or 'what is known' in Latin.The basic building block of data is a character which is the combination of zeros and ones in either American Standard Code for Information Interchange (ASCII, 7-bits) or Extended Binary Coded Decimal Interchange Code (EBCDIC, 8-bits). This ASCII or EBCDIC in form of zeros or ones is the language that computers understand. Table 1 shows rudimentary ASCII codes. For example if you type 2Ax#, it will be 50 for 2, 65 for A, 120 for x and 35 for #. The corresponding binary codes in memory by combining zone with bit, would be 0110 010 for 50, 1000 001 for 65, 1111 000 for 120 and 0100 011 for 35. This is how the numbers, symbols and letters that we type, either through computer or mobile phone, would appear in the memory. The character consists of upper and lower cases letters (Aa, Bb,…,Zz), numeric digits (0,1,2,…,9) and symbols (,..?, +, &, %, N, etc)

Table 1: Binary codes of ASCII in memory

zone/code	000	001	010	011	100	101	110	111
0000								
0001								
0010								
0011								
0100				# (35)	$ (36)	% (37)	& (38)	
0101			*(42)	+(43)	,(44)	= (45)	. (46)	
0110	48 (0)	49 (1)	50 (2)	51 (3)	52 (4)	53 (5)	54 (6)	55 (7)
0111	56 (8)	57 (9)						
1000		65 (A)	66 (B)	67(C)	68 (D)	69 (E)	70 (F)	71 (G)
1001	72 (H)	73 (I)	74 (J)	75 (K)	76 (L)	77 (M)	78 (N)	79 (O)
1010	80 (P)	81 (Q)	82 (R)	83 (S)	84 (T)	85 (U)	86 (V)	87 (W)
1011	88 (X)	89 (Y)	90 (Z)					
1100		97 (a)	98 (b)	99 (c)	100 (d)	101 (e0	102 (f)	103 (g)
1101	104 (h)	105 (i)	106 (j)	107 (k)	108 (l)	109 (m)	110 (n)	111 (o)
1110	112 (p)	113 (q)	114 (r)	115 (s)	116 (t)	117 (u)	118 (v)	119 (w)
1111	120 (x)	121 (y)	122 (z)					

A combination of letter, digit and symbol is called alphanumeric (#2B, N2.50k). A computer can accept alphanumeric and store them in memory in form of zeros and ones. A combination of zeros and ones forms a character and the combination of characters forms datum or fact. Facts or data are put together to form a record. Records are combined together to make a file. Files are combined together to make database. Once data have been processed by the computer, they are called information. Figure 1 illustrates the chain of forming a database.

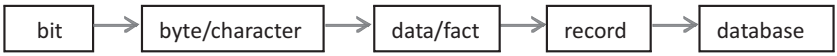


Figure 1: Chain of database

Data can also take the form of photograph, sound, video, music and display elements. These forms of data are represented in computer with different data types. Some of the common types of data include the following:

i a single character
ii Boolean (true or false)
iii text (string)
iv number (integer or float point)
v picture
vi sound
vii video

1.2 What is database?

Data base is a collection of information that exists over a long period of time often many years which is managed by a database management system.

According to Daniel Martin, database is a collection of data that obeys three criteria;

a exhaustivity
b non-redundancy
c appropriate structure

Exhaustivity means that the data about the subject are actually

present in the database.
Non-redundancy means that each individual piece of data exists only once in database.
Appropriate structure means that data are stored in such a way as to minimize the cost of the expected processing and storage. Database today is essential to every sector: agriculture, economy, business, mining, etc.
Whenever you visit a major website such as Google, Yahoo, Amazon or thousands of smaller sites that provide information, there is a database behind the scenes, serving up the information you request. Many people interact either directly or indirectly with database systems through credit card statement, bank tellers, airline reservation agent, etc. The Automated Teller Machine (ATM) came along and lets users interact directly with databases. Phones interface to computers, allowed users to deal directly with databases. Phone could be used by the students to register courses in a University. The Internet revolution of the late 1990 sharply increased direct user-access to databases for example, when you access an online bookstore and browse a book, you are accessing books stored in a database, when you enter a request online, your request is stored in a database, when you access a bank website and retrieve your bank statement and transaction information, the information is retrieved from the bank's database system.
Accessing database forms an essential part of almost everyone's life today. The power of database came from a body of knowledge and technology that had developed over several decades and is embodied to specialized software called a Database Management System (DBMS). A database management system is a powerful tool for creating and managing large amount of data efficiently and allowing it to persist over long periods of time safely.

1.3 Reasons for Database

Before the introduction of database management system, organizations stored information in file processing systems. Keeping organisational information in a file processing system has

a number of major disadvantages. Consider a program to transfer five thousand naira only (₦5,000=00) from the account balance of Mr. Ade to the account balance of Mr. Babatunde. If a system failure occurs during the execution of the program, it is possible that the ₦5,000=00 was removed from the balance of Mr. Ade but was not credited to the balance of Mr. Babatunde, resulting in an inconsistent database state. Clearly, it is essential for database consistency that either both the credit and debit occur or that neither occur. The transfer of ₦5,000=00 must be atomic. The transfer must happen in its entirety or not at all. It is difficult to ensure atomicity in a conventional file processing system.

In file processing system, if the number of application progress to share the same data increases such situation might lead to interface explosions. Figure 2 illustrates interface explosions of file processing system.

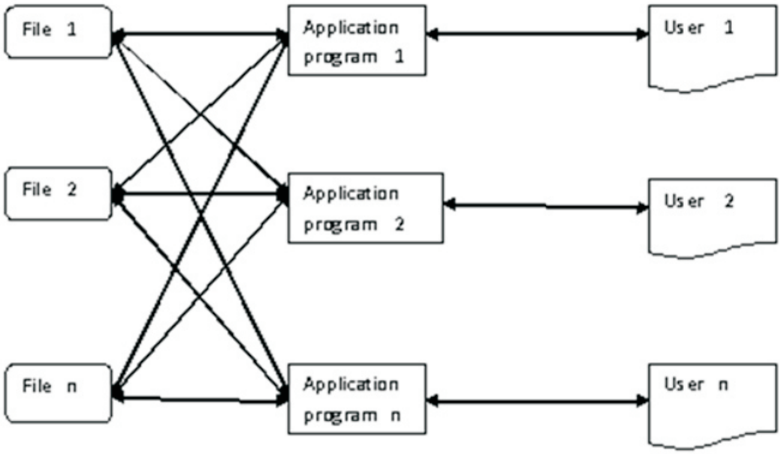


Figure 2: Interface explosion file processing system

It should be apparent that allowing different applications to share data in a traditional file processing environment can cause considerable problems simply because of the number of interfaces

required. It is noted that file processing system does not generally guarantee that data cannot be lost if it is not backed up and they do not support efficient access to data items whose location in a particular file is not known. File processing system does not directly support the user's ability to query data or modify the data. It does not support the recovery durability of the database in the face of failure errors of many kinds or intentional misuse. File system will not prevent situations such as two users modifying the same file at about the same time so that changes made by one user fail to appear on the file. For example, what would happen if in the banking system, maintaining account during system failures allows the money in the account to disappear? These difficulties among others prompted the development of database system.

1.4 Relationship of database, database management system and application programs

As indicated in Figure 3, application programs written in a high-level programming languages such as C++, C, JAVA, Python and Pascal access database through database management system such as MySQL, SQL, Microsoft Access, Dbase, Oracle, etc. In other words, database management system serves as the gatekeeper for database and application programs.

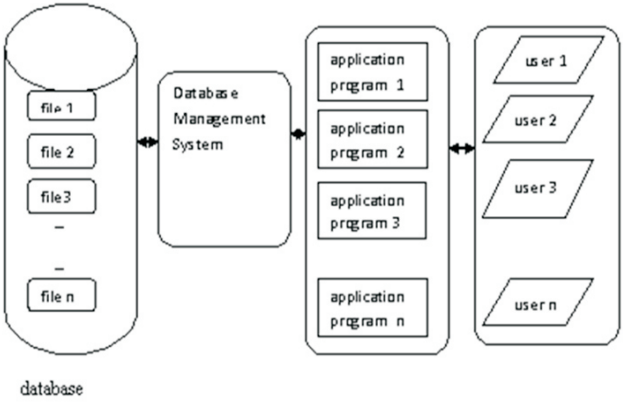


Figure 3: Relationship among database, DBMS and application programs

All access to the data in the database is performed through a database management system which understands and manipulates the logical data structures in the file.

Since all the application programs interface only with DBMS, they (application programs) require only a single interface to data in database. The interface of application programs to data in database can be at a logical rather than a physical level. It is not necessary for any application programs to know how a particular data item is stored as long as DBMS can provide the data in the form required by the application. For example, if the date of birth is stored as an integer, this does not matter to the application program as long as it is supplied in the format it requires. The property of needing to know nothing about the physical storage of the data by the application program is termed as data independence.

The only fear is that organization put all the data in a single basket, if anything happens to the basket then that is the end of data in database. This aspect of not losing the data in database together with other problems are parts of what computer scientists have been trying to solve in their research works.

1.5 History of database

Herman Hollerith was an American businessman, inventor, and statistician who developed an electromechanical tabulating machine for punched card to assist in summarizing information and later in accounting. Hollerith's idea for using punched cards from data processing came after he had seen how punched card was used to control Jacquard looms. Jacquard, a weaver who worked in France around 1810 originated the idea of using holes punched in card stock to control the pattern in a loom weaver. The use of punched cards in the Jacquard loom also influenced Charles Babbage who decided to use punched card to control the sequence of computations in his proposed analytical engine.

Hollerith's invention of the punched card tabulating machine patented in 1884 marks the beginning of the era of mechanized

binary code and semi-automated processing system. He founded a company that was amalgamated in 1911 with several other companies to form the computing tabulating recording company. In 1924, the company was renamed International Business Machines (IBM). Hollerith is regarded as one of the Seminal Figures in the development of data processing. He developed a mechanism using electrical connections to increment counter recording information. A key idea was that a datum could be recorded by the presence or absence of a hole at a specific hole location on a card. Hollerith determined that data specified locations on a card arranged in rows and columns, could be counted or sorted electromechanically. This was the first wholly successful information processing system to replace pen and paper. Punched cards were widely used as a means of entering data into computers. Data could also input from punched card decks and output to printers.

In 1950 and early 1960, magnetic tapes were developed for data storage. Data processing tasks such as payroll were automated with data stored tapes. Processing of data consisted of reading data from one or more tapes and writing data to a new tape. Tape and card decks could be read only sequentially and data sizes were much larger than main memory. Data processing programs were forced to process data in a particular order by reading and merging data from tapes and card decks. Every time a field was added or changed, all the programs working with the file would need to be re-written. Applications were batch-oriented because searching a tape to find or update a particular record was too slow to be practical. Master files were periodically updated with accumulated data or read through to produce reports.

The late 1960 and 1970 witnessed the use of hard disks. Hard disks allowed direct access to data. The position of data on disks was immaterial since any location on disk could be accessed in just tens of milliseconds. Data were freed from the tyranny of

sequentiality, with the hard disk, a computer could theoretically apply updates one at a time as new data came in and generate report as needed based on current data. Solutions such as hashing, linked lists, chain, indexing, inverted file, etc from algorithms were quickly devised. The concept of a database was made possible by the emergence of direct access storage media of magnetic disks which became widely available in the middle of 1960. In this period, Computer Science was just beginning to emerge as an academic field, but its early starts focused on programming language design, theory of computation, numerical analysis, operating system design and algorithm. Contrary to the academic neglect, the efficient and flexible handling of large collections of structured data was the central challenge. Data such as name, phone number, address, score, etc were considered the leftovers of processing information. Computers were just starting to become commercially available and when business people started using them for real world purpose, the leftover data suddenly became important.

Many business people are under false impression that designing and implementing database is a simple task that administrative staff can easily perform. In reality, designing and implementation a very good database is a huge challenge that requires analysis of an organization's need with careful design and implementation. The way database was designed and solved traditional paper-based information storage has evolved many stages which must be strictly followed.

1.6 Stages of database design

The earliest system designs of database were called the hierarchical and network model. The hierarchical model represented data as records which are connected with links. Each record has a parent record starting with the root record. Records in the hierarchical model contained one field. To access data using this model, the whole tree had to be traversed. These types of

database still exist today and do have a place in development despite the significant advancement in the technology. Figure 4 illustrates hierarchical database model. They are for example, used by Microsoft in the windows registry and in the file systems. They have advantages over more modern database models in terms of speed and simplicity. However, there are also many disadvantages one of them is that it is not easy to represent relationships between types of data. This disadvantage can be corrected through quite complex methods. The hierarchical database did solve many of the problems of paper-based system of colossal paper journals, ledgers, filing documents, etc.

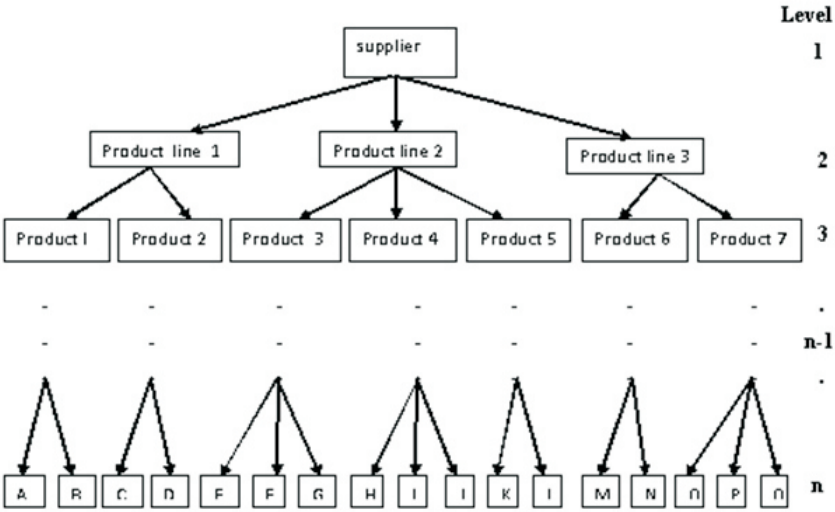


Figure 4: Hierarchical database model

Records could be accessed almost instantaneously. It had a full backup and recovery mechanism that meant the problem of lost data on file, due to damages, was a thing of the past.

In 1969, Computer Scientists of Committee of Data System and Language (CODASYL) released a publication that described the network model. It was the next significant innovation in databases.

It overcomes the restrictions of the hierarchical model. As shown in Figure 5, this model allows relationships and it has a schema that shows a diagrammatic representation of the relationships like in real life. The main difference between hierarchical and network model is that the network model allows each record to have more than one parent and child records. The network database model did improve on the hierarchical database model but it did not become dominant. The network and hierarchical models were used throughout the 1960s and 1970s because they offered better performance. Charles Bachman designed the first computerized database in the early 1960 which was known as the Integrated Data Store (IDS).

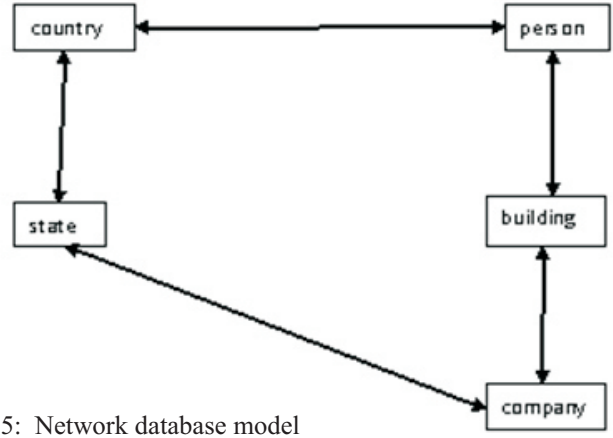


Figure 5: Network database model

He created IDS as a practical tool not as an academic research project. He designed IDS to meet the needs of an integrated system project called manufactory information and control system. The IDS stored and manipulated meta data about the record and their relationships. The IDS was designed to be used with a high level programming language of Common Business-Oriented Language (COBOL). This provided a measure of what is now called data independence for application programs. If a file was restructured

to add fields or modify their lengths, then the programs using it would continue to work properly. Files could be moved around and records could be reorganized without re-writing application programs against the same database. The IDS also included its own system of paging data in and out of memory to create a virtual memory capability transparent to the application programmer. The IDS used network database model where application programs could find records by one of the three methods:

1. the use of a primary key typically implemented by hashing.
2. navigating relationships called sets from one record to another
3. scanning all the records in a sequential order

The CODASYL approach, which is the network database model, was very complicated system and required substantial training. The IDS was shortly followed by the Information Management System (IMS), a database created by IBM in 1966. The IMS remained in use as of the year 2014.

Edgar Frank Ted Codd was an English Computer Scientist who worked for IBM. He was unhappy with the navigational model of the CODASYL approach, (IDS, and IMS) notably the lack of a 'search' facility. In 1970, Codd wrote a number of papers that outlined a new approach to database construction that eventually culminated in the ground breaking of the paper titled "A relational model of data for large share data bank". In his paper, he described a new system for storing and working with large databases. Instead of records being stored in some sorts of linked list of free-form records as in CODASYL, Codd's idea was to organize the data as a number of tables. Each table was used for different types of entity. Each table would contain a fixed number of columns containing the attributes of the entity. One or more columns of each table were designated as a primary key by which the rows of the table could be uniquely identified. Cross references between tables always used these primary keys rather than disk addresses.

Queries would join tables based on these key relationships using a set of operations based on the mathematical system of relational calculus for which the model takes its name. Splitting the data into a set of normalized tables or relations aimed to ensure that each fact was only stored once, thus simplifying update operations. Virtual tables called views could present the data in different ways for different users but views could not be directly updated. Codd's relational database model supports mathematical set of operations like union, intersection, difference, and Cartesian product. The model also supports select, project, relational join and division operation.
Codd used mathematical terms to define the model; relations, tuples and domain rather that tables, rows and columns. The model also enabled queries operations to be defined in terms of the established discipline of first-order predicate calculus because these operations have clear mathematical properties. An example could be a customer table which could include the following fields:
customer(cust-id, fname, lname, street, address, country, post-code, mobile-phone, e-mail)

First, the data type of each field is predetermined. This would assist in determining how to sort data. These tables/relations can have relationships such as one-to-one, one-to-many and many-to-many in a relational database. Figure 6 illustrates relational database model. These allow the relational database designs to show how one table relates to another. For example, a customer will buy many products. A graduate could use his certificate to teach or consult or perform tutorial classes from one house to another. This is a one-to-many relationship. The main advantage of the relational model is that it provides consistency in the data. The model implements a set of constraints, and these ensure the database functions as intended. The implementation of the relational database system met four properties as defined by Codd: ACID

A = Atomicity, C = Consistency, I = Isolation and D = Durability.
Atomicity ensures that the data operations will complete either with success or with failure.
Consistency means that if we perform any operation over the data, its value before and after the operation should be preserved. For example, the account balances before and after the transaction should be correct. The account balance should be conserved.

Student file

s/n	matric_No	name	dept	score
1	2001/0466	Ojo	csc	76
2	2001/0478	Bello	emt	66
3	2001/0368	Bala	ann	54

Course file

matric_No	Course_Title	Course_Code
2001/0368	Database system	csc407
2001/0478	Use of computer	csc701

Figure 6: Relational database model

Isolation means that there can be concurrent users for accessing data at the same time from the database. Thus, isolation between the data should remain isolated. For example when multiple transactions occur at the same time, one transaction effects should not be visible to the other transactions on the database.
Durability means that once database completes its operation and commits the data, data changes should remain permanent.
Modern business still use relational database to record their day-to-day activities and to help them make critical strategic decisions.

The IBM had invested heavily in the IMS model and was not interested in Codd's ideas. In 1973, Michael StoreBraker and Eugene Wong at University of California, Berkeley picked up Codd's paper and did research on relational database system which resulted into interactive

graphics and retrieval system (INGRES). They demonstrated that a relational database model could be efficient and practical. The INGRES worked with query language known as QUEL. This pressured IBM to develop Structured Query Language (SQL)/ Database System (DS) in 1974 which eventually replaced QUEL as the more functional query language. In Sweden, Codd's paper (1975) was also read and MySQL was developed from the middle of 1970 at Uppsala University. This software MySQL was the first significant database of the Internet and continues to be used by companies like Google, Facebook, Twitter, Flacker and Youtube.
In 1976 entity-relationship was proposed by P. Chain. This model made it possible for designers to focus on data application, instead of logical table structure. In 1980, SQL became the standard query language. The SQL allows an application to gain access to the data needed by a user. It can either retrieve all the data from a table or even a database, or just one individual field determined by a set of criteria. For example, an application may only require the name of a Professor associated with a particular course and they may not need any more data from the tables.

In early 1990, ORACLE database management system was developed by IBM. In the middle of 1990s, Internet came out which led to exponential growth of the database industries. Average desktop users began to use client-to-server database system to access computer systems that contained legacy of data. There was an increased investment in online business which resulted in a rise in demand for Internet database connectors such as Front Page, Active Server Page, JAVA Servelets, Dream Weaver Cold Fusion, Enterprises JAVA Beans and Oracle Developer 2000. The use of cgi, gcc, mysql, Apache and other systems brought upon source of solution to the Internet. With increased use of point-of-sales technology, online transaction processing and online analytic processing system began to come of age. Database system now had to support very high transaction processing rates, as well as very high reliability and 24 by 7

availability of no down-time for scheduled maintenance activities. Database system also had to support web interface to data.

Not only Structured Query Language (NoSQL) came out about as a response to the Internet and the need for faster speed as well as processing of unstructured data. The NoSQL used a distributed database system. This NoSQL used an ad-hoc method of organizing data and processes high volumes of different kinds of data. The widespread use of NoSQL can be connected to the services offered by Twitter, Linkedln, Facebook and Google. Each of these organizations store and processes colossal amounts of unstructured data which must be adequately protected. The NoSQL database can provide high performance at an extremely large scale and never shuts down. In fact, there are four kinds of NoSQL databases with each having specific qualities and characteristics. These kinds of NoSQL are:

i document stores
ii common stores
iii key-values stores
iv graph data stores

Document stores : document stores or document-oriented database manages, stores and retrieves semi-structured data. Documents can be described as independent units that improve performance and make it easier to spread data across a number of servers. Document stores come with powerful query engine and indexing controls that make queries fast and easy.

Examples of document stores are Mongo database and Amazon database.

Column stores: This stores data as portions of column instead of as rows. This changes the storing of data in focus from row to a column. Column databases maximize their performance when large amounts of data are stored in a single column.

Key-value stores: Key-value stores use primary key to access database. Examples are Riak, Berkeley DB and Aerospike.

Graph data stores: These databases are based on graph theory and

work well with data that can be displayed as graphs. They provide a very functional of Big Data. Examples of graph databases are Neo4j, GraphBase, Titan, etc.

Many NoSQL systems run on nodes and large clusters. This allows for significant scalability and redundant backups of data on each node. It uses different technologies at each node. The first half of the 2000s saw the emerging of extensible mark-up language (XML) and the associated query language of XQuery as a new database technology. The XML is widely used for data exchange as well as storing certain complex data types. Several novel distributed data-storage systems have been built to handle the data management requirement of very large web sites such as Amazon, Facebook, Google, Microsoft and Yahoo. These websites are offered as web services, websites that can be used by application developers.

Database has allowed companies to develop incredibly sophisticated enterprises resource system that gather data from every part of an organization and store it all in a central database. Data is collected from factories, offices, remote workers, sensory and anywhere they are useful. Executive and managers can now see real-time information of their organization and can use this to help them understand more about the decision they need to make. Systems have to be designed to provide the right information to the right people at the right time. Database also allows organizations to work more effectively with their customers and suppliers. Without database, we would still be losing valuable information and the digital revolution would not have happened. The coming industrial revolution also called industry 4.0 will be driven by data and it will transform the lives to every customers and business in the world. Today, database is the 5th utility after water, electricity, gas and telephone and 3rd in Information Technology after Personal Computer and Internet.

Today, there are databases that store petabytes (10^{15} bytes) of data and serve it all to users. Google holds petabytes of data and it was

not stored in DBMS but in specialized structures optimized for search engine queries. Satellites send down petabytes of information for storage in specialized systems. Storing a picture typically takes much more space. Databases such as Flick, store millions of pictures and support search of those pictures. Even a database like Amazon has millions of pictures of products to serve. Pictures still consume space. Movies consume much more. An hour of video requires at least gigabytes sites such as Youtube which holds hundreds of thousands or millions of movies and made them available easily. Data mining techniques are now widely deployed; examples of applications include web-based product recommendation systems and automatic placement of relevant advertisements on web pages.

Presently, we have new SQL which is a class of modern relational database that aims to provide the same scalable performance of NoSQL system for online transaction processing (read-write) workloads while still using SQL properties of ACID. In today's world of competition, information, which is stored in database, is the key to success. Availability of right information from the database at the right time can make all the difference. Today, relevant information from database outweighs the price of gold. Internet and World Wide Web which are bigger and better database have changed the way we work, learn and stay in touch with others. We can purchase almost anything without wasting time and money. We can purchase almost anything without physically travelling to a store. We can quickly locate products and services via Internet that may not be available in our geographic neighbourhoods. This happens due to powerful database: bigger and better database for innovative advancement in industrial revolution.

Companies use bigger and better database in their systems to improve operations, provide better customer service, create personalized marketing campaigns and take other actions that, ultimately, can increase revenue and profits. Thousands of data are

created and used daily every second. The real life examples are communication data we purchase from telecommunication providers.

Bigger and better databases are mainly for transactions in Banks, Government, Industries, Telecommunication providers, such as MTN, Glo, Airtel and etc. These made room for a bigger database due to the geometric increment in the volume of information produced by users to be stored in the database daily. However, the reliability, efficiency, and effectiveness due to these increases must be guaranteed. Infact, it has been discussed by some schools of thought that the basic needs of life- such as good, shelter and clothing are not enough without information, as we are in information age. Meanwhile, there is need for a better database to accommodate these sets of ever-increasing information which must be accurate, reliable, secured, etc. Furthermore, for a global innovative advancement in all facets of life, bigger and better databases are necessary as an ENGINE that propels any innovative drive for proper record storage and retrieval for a quality decision making process in the era of industrial revolution with the support of Artificial Intelligence, Machine Learning, Internet of Things, etc. This Fourth Industrial Revolution creates a world in which virtual and physical systems of manufacturing cooperate with each other in a flexible way at the global level. It is however, not only about smart and connected machines and systems. Its scope is much wider. Occurring simultaneously are waves of further breakthroughs in areas ranging from gene sequencing to nanotechnology, from renewable energies to quantum computing. It is the fusion of these technologies and their interactions across the physical, digital and biological domains that make the Fourth Industrial Revolution fundamentally different from previous revolutions and the engine that will drive it successfully is bigger and better databases.

1.7 Forms of databases

There are many forms of databases both structured and unstructured, that have been designed for specific data. Among them include: an in-memory database, active database, a cloud database, database warehouse, a deductive database, distributed database, embedded database, end-user database. document-oriented database, federated database, graph database, multi-database, array database, hypermedia database, knowledge database and mobile database. Other forms are operational database, parallel database, probabilistic database, real-time database, spatial database, temporal database, terminology-oriented database and unstructured database.

1.8 Types of database models

A database model is a type of data model that determines the logical structure of a database and fundamentally determines in which manner data can be stored, organized and manipulated. Common logical data model for database include flat database model, hierarchical database model, network database model, graph database model, relational database model, entity-object oriented database model, entity-relationship database model, enhanced entity relationship database model, document oriented database model, entity- attribute-value database model, cloud database model, not only sql database model, distributed database model, centralized database model, enterprise database model, personal database model, operational database model but also star schema database model. Other database model are associative database model, multidimensional database model, multi-value database model, XML database model, semantic database model, content store database model, event store database model and time series database model.

1.9 Architecture of database management system

Database architecture focuses on the design, development, implementation and maintenance of computer programs that store and organize information for businesses, agencies and institutions. Database architecture develops and implements software to meet the needs of the users. The design of a DBMS depends on its architecture. It can be centralized or decentralized or hierarchical. The basic client/server architecture is used to deal with a large number of PCs, web servers, database servers and other components that are connected with networks. The architecture of a DBMS can be seen as either single tier or multi-tier. The tiers are classified as follows:

1. **1-tier architecture**
2. **2-tier architecture**
3. **3-tier architecture**
4. **n-tier architecture**

For example, one-tier architecture involves putting all of the required components for a software application or technology on a single server or platform.

1.10 Challenges of database management system

There are many relational databases from which many companies are finding it difficult to choose and evaluate. There is a limit on scalability of database management system as other software programs of operating system and configuration of hardware have limitation on resource usage. The increase in data volume is another challenge to database system. Research shows that we have created more data in the past two years than in the entirety of the human race. Data security is another big challenge in the field of database management system. Decentralization of database management system is also a challenge. The question is "how would the data be distributed?" which is still a big challenge?.

1.11 Database in Nigeria

There are quite numbers of database in Nigeria such as Independent National Electoral Commission (INEC), National Identity Management System (NIM), Federal Road Safety Commission (FRSC), Nigeria Banking System (BVN), Nigeria Telecommunication Operator, National Youth Service Corps (NYSC), etc. These databases in Nigeria are not unified. The data in each of these databases are not aware of the other data. For example, Mr Ade who lived in Bodija Street, Ibadan, Oyo State, according to his National Youth Service Corps may live in Challenge Street in Lagos according to his Bank Statement (BVN database). It is very difficult to trust data in Nigerian database. Database of Federal Road Safety Commission captured records of all persons capable of driving on Nigerian roads. The INEC database always updates the records with miss-matched surnames, date of birth, place of birth and wrong addresses which are not in line with the principle of database. During coronavirus pandemic in Nigeria, it is difficult to distribute palliative food to people as it resulted into duplication where some people might register in two or three states within a short period of time. Today it is difficult to determine whether some herdsmen are actually Nigerians since there is no centralized database to capture them. It is not possible to use database in Nigeria to plan for either economic growth or national planning. The introduction of biometric into National Identification Management System could serve as a central database management system for the nation. This would play much broader roles in driving socio-economic development and security of the nation.

2.0 Contribution to knowledge

A few of findings from my personal and collaborative research works is presented in this section. In 1993, effort was made to design accounting system for processing financial interests of jointly owned companies which would show the capital contributed and commitment by participants, as well as how these

capitals are utilized in running the business. The system itemized accounting items owned by parties for the purpose of up-to-date records and safeguarding them. The system showed the company's liabilities, contingencies funds and expenses incurred by both investor and venture in relation to their interests. Designing these systems digests the solution of concatenation of individual components of accounting items in jointly owned companies into one system. Emphasis was put on documentation of these systems with respect to file design, input used and final outputs for decision making. The system design collected business transaction and arranged them in their specific master files. Information in every file was processed step-by-step and could be easily modified during inflation and changes in certain accounting rules and regulations. Output files described accounting items of venture and investor which could be used to evaluate financial strength and performance of the business. The implementation of the ideas incurred could improve the accounting process in the specific conditions of jointly owned companies(Adio Akinwale, 1993).

Later on, a decision support system was built to evaluate the financial strength and profitability of joint venture companies by employing untranslated and translated statements using temporal and current methods. The interpretation of translated statement against untranslated gave a fuller picture of the joint venture financial position and performance. The information in translated statement is more reliable than information in the untranslated statement using temporal method(Adio Akinwale, 1994).

In another effort, fragmentary knowledge base for translated statement operation was carried out in rule forms. These rules were analysed, programmed and stored by using macro programming in Quattro without repetitive. The program allowed both ventures and investors to keep track of their financial rules and use the presented information for interactive decision making (Adio Akinwale, 1994).

Error back propagation algorithm and regression analyses were also used to analyse and predict untranslated and translated Nigerian Stock Market price. A zero mean unit variance transformation was used to normalize the input variables in order to allow the same range which makes them to differ by order of magnitude. A 5-j-1 network topology was adopted because of five input variables in which variable j was determined by the number of hidden neurons during network selection. The untranslated and translated normalized Nigeria Stock Market price served as input into both error back propagation algorithm and regression model which were written in JAVA Programming Language. Simulation was performed and the results were analysed and compared. The performance of translated Nigeria Stock Market price using regression analysis or error back propagation was more superior to untranslated Nigeria Stock Market price, even in term of percent accuracy(Akinwale *et al*, 2005).

An attempt was made to use semantic schema model to ease and facilitate query formulation as against relational databases. A mix-fix function was employed to derive query paths that produce the desired results. Different modules were written to display user-friendly menu and convert conceptual query language syntax to structured query language form. This approach allows end users to formulate query in a more user-friendly format. It also allows formulation of queries that entail more than two entities; thereby, avoiding creation of virtual relations which normally occupy large spaces in database environment and a very fast query processing time (Arogundade and Akinwale, 2003)

A generalized assignment method and extended conjugate gradient method were used to manage dynamic storage allocation in distributed database systems. They viewed different node storage capacities with arbitrary host memory capacities and generate remarks. The work concluded that extended conjugate

gradient method is more efficient than generalized assignment method as an optimization technique in managing dynamic storage allocation in distributed database systems (Adekoya and Akinwale, 2003).

The front page and direct authoring with hypertext mark-up language (HTML) were used to design web pages as they relate to world environment. The two web methods generate codes which produce the same results. Results were evaluated and recommended tools were proposed based on the results. We found that it is desirable to design web page on direct authoring with HTML (Akinwale and Akanmu, 2005).

Effort was also made to determine functionalities that could be incorporated into a database management system that operates in an open network system environment. These functionalities include application of open password for open database system, creation of register database, creation and monitoring of activity log file, creation of privacy lock on schema and application of a locking control on file. A system was designed and implemented with sample database which provides security for open database system by the desired users(Akinwale *et al*, 2009).

Multi-level cryptographic functions for the functionalities of open database system were proposed as open password entry with active boxes and agent. The design and implementation would allow users to have free access into open database system. A user entering his password only needs to carefully study the sequence of code and active boxes that describe his password and then enter these codes in place of his active boxes. Integrating this scheme into an open database system is viable in practice in term of ease of use and improved security level of information(Akinwale *et al*, 2011).

A research was also carried out to select three good hash functions which uniformly distribute hash values that permit their internal states and allow the input bits to generate different outputs. These hash functions were used in different levels of hash tables that are coded in JAVA Programming Language and a quite number of data records serve as primary data for testing the performance. The results showed that two-level hash tasks with three different hash functions gave a superior performance over one-level hash table with two hash functions or zero-level hash table with one function in terms of reducing the conflict keys and quick look-up for a particular element. The results assisted to reduce the complexity of join operation in query language from $O(n^2)$ to $O(1)$ by placing larger query result, if any, in multi-level hash tables with multiple hash functions and generate shorter query result(Akinwale and Ibhralu, 2009).

Recent research development on egocentric social network analysis focuses on the causes and consequences of the relationships among sets of people rather than on the features attribute data of individual and their basic database operations. An effort was made to describe a record type of network data model that would be suitable to perform database operations in social network analysis. The research work described two schemas that would be needed for web space interface and database operations. A force directed algorithm was used to display membership vertices that show link of membership number, name and pictures. A breadth first search algorithm was embedded to avoid endless searching in the cycled nodes. Measurement of degree of centrality, betweeness and closeness was incorporated to determine the relationship among the nodes and their linkages. Implementation of database operations into social network analysis reveals not only social information but also critical economic reports of the individuals. Observation of the work showed that users can impose any relevant query related to the defined schemas and the results would be available to motor both social and economic activities of

the memberships (Akinwale *et al*, 2010).

Designing normalized relation is subject of great difficulty, yet of great importance. Effort was made to present automatic derivation of database keys as a step forward solution to designing normalized relations. The research work studied relational schemas and their functional dependencies to derive keys of primary, candidate, alternate and super key of relational database. The necessary algorithms to derive database keys were analysed, modified and coded in JAVA Programming Language to display graphical key derivation input interface. The input interface accepts any length of schemas' attributes and functional dependencies to compute power set, f-closure, derived keys and specific group of keys which can aid database designers to control anomalies at initial stage of database design. The system also assists the students to find, see and learn database key derivation. The effectiveness of the system has a positive impact on the student's performance on normalization technique. Figure 7 shows the interface of the implementation of the automatic key derivation.

Figure 7: Automatic generation of database keys

These keys are very important in the process of normalization, decomposition, loss less join and dependency preserving. For example, determine candidate keys for a small relation schema of R(A, B, C)
with FD's A→B,
B→C is easy
while determine candidate keys for a large relation schema of
R(A, B, C, D, E, F, G, H, I, J, K, M, N, P)
with FDs: A, G →B, C, D, E, F, H, J, K, L, M, N, P
J →K ,L, M, N, P
M →N, P
G →H
is very difficult

The condition of decomposition says that decomposition relation R onto R_1 and R_2 is loss-less if and only if one of the following dependencies is in F^+

$(R_1 \quad R_2) \quad (R_1 - R_2)$
$(R_1 \quad R_2) \quad (R_2 - R_1)$

This means that attributes R_1 and R_2 must contain candidate key R_1 or R_2 as my research has proved.

Application of loss-less join algorithm for each normal form and reverse the decomposed relations without errors were also established in my research. The work also resolves the problem of join operation

Traditional method is
$R_1 \text{ x } R_2 \text{ x } R_3 \text{ x} \ldots \text{x } R_n$
This method is expensive and time consuming.
For example
select attribute_1, attribute _2, ... , attribute _n,
from R_1, R_2, ..., R_n
where (R_1. attribute _1 = R_2. attribute _2) and
(R_3. attribute _3 = R_5. attribute _5)

The problem is which key must be used to perform the join operations. Is it primary key or alternative key? A visualization tool for teaching and learning decomposition of relational database schema has resolved the following problems:

Determination the database keys
Normalization schemas
Decomposition relations when there is a violation
Loss-less join after decomposition
Dependency preserving on decomposed relations
Join operation of two or more relations

These above mentioned problems, if they are not properly resolved, would lead to poor database design, anomalies and poor matching which have been addressed in my reearch contributions (Akinwale *et al*, 2011; Akinwale and Arogundade, 2012).

A framework for reducing multi-dimensional database to two dimensions was proposed using matrix decomposition algorithm. It employed decomposition algorithm based on Boyce Codd normal form to minimizing anomalies. The decomposition and reversible algorithm were used on relationship among object attributes and were implemented. The implemented program ran on sample genetic epistasis datasets of up to 10 dimensions and it was shown that multi-dimensional datasets could be reduced to two dimensions. It was established that the time taken to generate a sequence of tuples from multi-dimensional database to a 2-dimentional dataset was directly proportional to the number of genes considered. The result showed that the reduced 2-dimensional database did not require any in-built functions which take long processing time for generating query result as against querying of multi- dimensional dataset. The reduced 2-dimension dataset was reversible to the original multi- dimensional dataset for lossless join operation which indicated that there was no loss of data values or tuple. The method was compared with existing reduction techniques and it was found that data access was very fast with decomposition algorithm than relational model. Figure 8 illustrates the performance of the two methods.

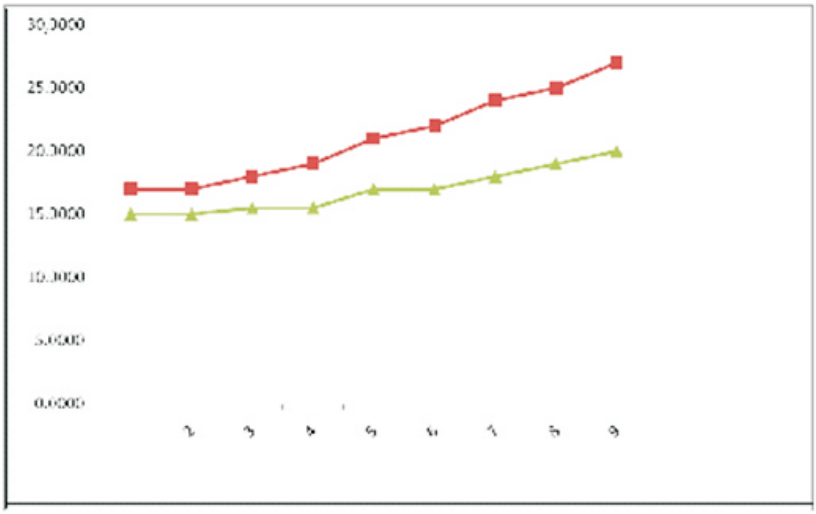


Figure 8: The performance of traditional join operations and matrix decomposition algorithm (Akinwale *et al*, 2011).

A hierarchical database model for querying economic network independent distribution of goods and services was carried out and implemented alongside B-tree and pre-order algorithm to insert and traverse participants' records for easy processing. N-level models were adopted to calculate each level and sub-level cluster commission. A user interface was designed for easy interaction with the system. Users can interact with the system by moving the mouse across the nodes at any level of hierarchical database model. At each node, users can view their positions on the hierarchical tree, as well as their due commission. A form is always displayed for the users to query the system for any decision making. The major problem is that the users must have basic knowledge of writing simple structural query language statement(Akinwale Adio Taofiki, 2011).

Text similarity proves to be useful in various tasks of string

processing system. N-gram methods have been proposed to locate exact matching in text processing applications. Generalized n-gram matching technique during substring processing tends to be time consuming and it occupies large spaces. These factors motivate to explore the features of generalized n-gram matching technique for electronic test that would provide low computational time; easier implementation and positively translate into effective usage. The new methods are expected to calculate high similarity values from the template pattern and matching texts with the minimum processing times and high rates of performance to price. This motivates for the development of new similarity measures which would overcome most of the unsatisfactory features of generalized n-gram methods.

Five new similarity measures of
bi-n-gram,
tri-n-gram,
set-based trigram,
oddgram and
sumsquare gram methods
have been proposed to improve the performance of generalized n-gram matching technique in electronic test at programming languages, subjective questions, case-based reasoning, medical database terminologies and word lists. The new methods where they are most applicable, assigned high values of similarity measure and performance to price while low values were also assigned to running times.

n-gram method
Similarity of two strings s_1 and s_2 can be determined via the n-gram method as follows:

$$sim_n(s_1, s_2) = \frac{1}{N-n+1} \sum_{i=1}^{N-n+1} h(i)$$

where h(i) = 1 if n-element subsequence beginning from position i in s_1 appears in s_2
h(i) = 0 otherwise;
N-n+1 = number of n-element subsequence in s_1.

Generalized n-gram matching for two strings
Generalized n-gram matching was introduced by Niewiadomski. The algorithm matches an answer string to a template string as follows:

$$sim(s_1, s_2) = f(n_1, n_2) \sum_{i=n_1}^{n_2} \sum_{j=1}^{N-n+1} h(i, j)$$

for (s_{1i}, s_{2j})
where

$$f(n_1, n_2) = \frac{2}{(N - n_1 + 1)(N - n_2 + 2) - (N - n_2 + 1)(N - n_1)}$$

denotes the number of possible substrings not shorter than n_1 and not longer than n_2 in s_1, h(i, j) = 1 if an i-element long substring of the string s_1 starting from j-th position in s_1 appears (at least) once in s_2 (otherwise) h(i, j) = 0 , if all substrings from one argument of comparison are found in the other, the final similarity degree is evaluated to 1 which is interpreted as the identity of s_1 and s_2 (Niewiadomski Adam, 2008).
$N(s_1)$, $N(s_2)$ = the length of string s_1 and s_2, N = max {$N(s_1)$, $N(s_2)$ },

Bigram model
Generalized n-gram matching is normally used to derive bigram where n is equal to 2, hence the function is as follows:

$$sim_n(s_1, s_2) = \frac{1}{N - n + 1} \sum_{i=0}^{N-n+1} h(i) = \frac{1}{N - 2 + 1} \sum_{i=1}^{N-2+1} h(i) = \frac{1}{N - 1} \sum_{i=0}^{N-1} h(i)$$

Bigram into Bi-n-gram

Bigram is modified into bi-n-gram as follows:

$$sim_n(s_1, s_2) = \frac{1}{2N-n+1}\sum_{i=0}^{2N-n+1} h(i) = \frac{1}{2N-2+1}\sum_{i=1}^{2N-2+1} h(i) = \frac{1}{2N-1}\sum_{i=0}^{2N-1} h(i)$$

Trigram model

Two strings s_1 and s_2 are determined via the n-gram method where n = 3 is called trigram, hence the function is as follows:

$$sim_n(s_1, s_2) = \frac{1}{N-n+1}\sum_{i=0}^{N-n+1} h(i) = \frac{1}{N-3+1}\sum_{i=1}^{N-3+1} h(i) = \frac{1}{N-2}\sum_{i=0}^{N-2} h(i)$$

Trigram into Tri-n-gram

Trigram is modified into tri-n-gram as follows:

$$sim_n(s_1, s_2) = \frac{1}{3N-n+1}\sum_{i=0}^{3N-n+1} h(i) = \frac{1}{3N-2+1}\sum_{i=1}^{3N-2+1} h(i) = \frac{1}{3N-1}\sum_{i=0}^{3N-1} h(i)$$

Oddgram method

Oddgram method was inspired by the generalized n-gram matching which takes n(n-1)/2 substrings for processing before measuring the performance. The oddgram would take half substrings of generalized n-gram matching for processing the performance which would still reduce the running time. The oddgram method is illustrated as follows:

$$\mu\eta_z(z_1, z_2) = \frac{1}{M}\sum_{i=1}^{M(z_1)} \overbrace{j = 1, ..., M(z_1)}^{\max} sim_{odd}(s_1, s_2)$$

where

z_1 = sentence for pattern matching of s_1,

z_2 = sentence for test matching for s_2

N = max {N(s_1), N(s_2) } which is the maximum length between string s_1 and s_2

If N is odd then for $N = \left\lceil \frac{N}{2} \right\rceil$

$$sim_{odd}(s_1, s_2) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N-i+1} h(i, j)$$

else

$$sim_{odd}(s_1, s_2) = \frac{1}{N^2 + N} \sum_{i=1}^{N} \sum_{j=1}^{N-i+1} h(i, j)$$

M = max {M(z_1), M(z_2) }

Sumsquare gram for two strings

Like the oddgram, sumsquare gram was inspired by the generalized n-gram matching in which processing time is quadratic for every n-gram in the query string of line statement. While similarity measures of n-gram are easy to generate and manage, they do require quadratic time and space complexity and therefore ill-suited to both oddgram and sumsquare gram which work in quadratic. Oddgram and sumsquare gram methods are expected to write their results into similarity measure (s) between a pair of submission (pattern matching and text matching). Given pattern matching and text matching i and j, s_{ij} will be near to 1 if both matchings are considered identical and near to 0 if they are very dissimilar. That is, oddgram and sumsquare grams are normalized to fall within the interval [0, 1]. Similarly, similarity measure of oddgram and sumsquare gram are expected to be symmetric, that is, the equality $s_{ij} = s_{ji}$ is expected to hold for every i and j.

The sumSquare gram method is as follows:

$$\mu\eta_z(z_1, z_2) = \frac{1}{M} \sum_{i=1}^{M(z_1)} \overbrace{j = 1, ..., M(z_1)}^{\max} sim_{Sq}(s_1, s_2)$$

where

z_1 = sentence for pattern matching of s_1,

z_2 = sentence for text matching for s_2

N= max {N(s_1), N(s_2) } which is the maximum length between string s_1 and s_2

$$N = \left\lfloor \sqrt{N} \right\rfloor$$

T = time-to-jump = N - 1
P = first-jump = $N^2 - (N - 1)^2$

$$sim_{Sq}(s_1, s_2) = \frac{6}{N(N+1)(2N+1)} \sum_{i=1}^{P} \sum_{j=1}^{T} h(i, j)$$

$M = \max \{M(z_1), M(z_2)\}$

Set-based trigram method
Set-based trigram method was inspired by the theory of set similarity measure as follows:

$$\mu(\phi) = 0$$
$$\mu(A \cup B) \leq \mu(A) + \mu(B)$$
$$A = B \; -> \mu(A) = \mu(B)$$
$$A \subseteq B -> \mu(A) \leq \mu(B)$$

It measures the similarity between two sets of entities. It is used to measure the similarity of two strings in terms of the number of common trigrams. It increases the weight of string sharing between pattern and text matching. It is asymmetric because it is not considered (false, false) to be a matched patterns. The method is described as follows:

$$\mu\eta_z(z_1, z_2) = \frac{1}{M} \sum_{i=1}^{M(z_1)} \overbrace{j = 1, ..., M(z_1)}^{\max} sim_{Set-based-tri}(s_1, s_2)$$

where
z_1 = sentence for pattern matching of s_1,
z_2 = sentence for text matching for s_2

$$set - based - tri(s_1, s_2) = \frac{3(trigram(s_1 \cap s_2))}{(trigram(s_1)) + (trigram(s_2)) + trigram(s_1 \cap s_2)}$$

where

$trigram(s_1 \cap s_2)$ is the number of character trigram found in both strings s_1 and s_2.
(trigram(s_1)) and (trigram(s_2)) are the number of trigram in both string s_1 and s_2 respectively.
N = max {N(s_1), N(s_2)},
M = max {M(z_1), M(z_2)}

Complexity time of the new methods
The complexity time of $O(n^2)$ for generalized n-gram grows larger than computational time of $\frac{O(n^2)}{2}$ for oddgram in respect to sufficient large n. The same case to the complexity time of $\frac{O(n^2)}{2}$ for oddgram grows larger than computational time of $\frac{O(n^2)}{\sqrt{n}}$ for sumsquare gram to sufficient large n. The execution time for computing the methods is insignificant with O(1) for bi-n-gram, tri-n-gram and set-based trigram while O(n) is for generalized n-gram, oddgram and sumsquare gram method. This indicates that the asymptotic behaviour of computing bi-n-gram, tri-n-gram and set-based trigram is faster than asymptotic behaviour of computing generalized n-gram, oddgram and sumsquare gram. Moreso, the asymptotic behaviour of computing generalized n-gram is slower than the asymptotic behaviour of computing oddgram of which asymptotic behaviour of computing sumsquare gram is faster than asymptotic behaviour of computing oddgram method. Any of these new methods which are able to compute exact matching between pattern and matching text, will score higher similarity values than those which cannot do so. Their evaluation would rely on a dataset D that contains a desire output.

The performance of each new method would be a function of real output D_0 for the desired input of

$$D : f(D, D_0) \rightarrow R.$$

This kind of evaluation is easily repeatable once D and f are given. It is straightforward to compute and evaluate the performance of existing methods with new ones. Thus, the performance of old and new methods with outputs would equal D_0 and D_{00} which could be compared as follows:

$$f(D, D_0) - > f(D, D_{00})$$

The performance of the new methods equals the performance of the system

$$g(Ds, \ D_0 \ s) \rightarrow R$$

where D_0 s is an output of the system. It was proved that the new methods perform comparably to the baselines of the given examples. Each new method has its pros and cons in term of text processing applications. One way to significantly improve over the baseline of the examples would be to use different datasets for testing the performance of each method along with the existing methods.

Five datasets that are used

1 similarity measure in word lists,
2 similarity measure in subjective questions,
3 similarity measure in programming language
4 similarity measure in case-based reasoning and
5 similarity measure in medical data base.

The performance of the new methods demonstrated that they have specificity and universality in the similarity measure of programming languages, subjective questions, case-based reasoning, medical database terminologies and word lists.

For example, two methods of bi-n-gram and tri-n-gram have been proposed to improve the performance of n-gram model of similarity measure in electronic test at programming languages. The new methods have increased the number weight letter by two times for each statement line in programming language codes

thereby allowing more string sharing. The effect of the modification permits similarity values of the two methods to be so closed to the expert judgment. The experimental results indicated that bi-n-gram achieved high performance with qualitative values and outperformed than other five methods. The results obtained from tri-n-gram in similarity values are very close to the results of experts which indicate that it can be successfully used in electronic test at programming languages. Figures 9, 10 and 11 depict the performance of bi-n-gram and tri-n-gram in electronic test at programming languages.

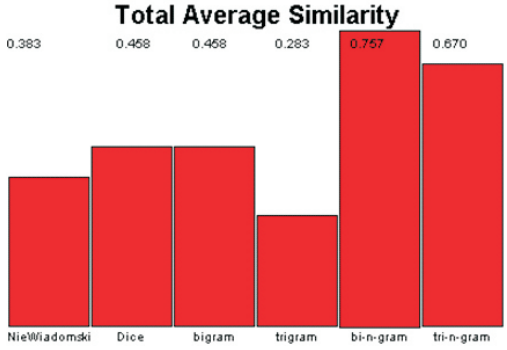


Figure 9: Total average similarity of the six mehtods

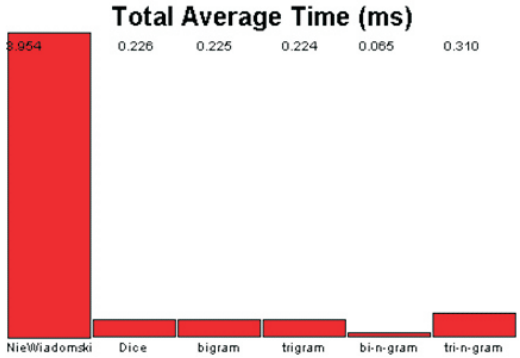


Figure 10: Total average running time of the six methods

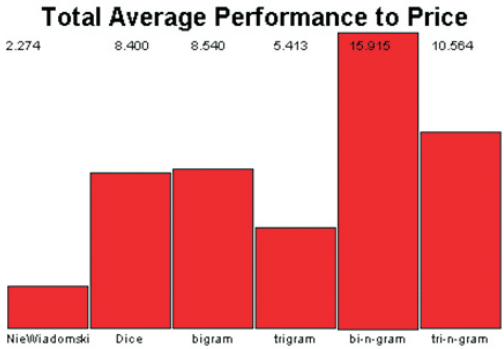


Figure 11: Total average performance to price of the six methods

In the area of medical database terminologies, oddgram, set-based trigram and sumsquare gram methods demonstrated a great improvement in the effective and efficient retrieval of relevant medical terms from the database to the users' requests. Figure 12 describes similarity testing using medical terms.

In the similarity measures of case-based reasoning, sumsquare gram and set-based trigram methods have the highest values of similarity measures. They produced low score values of highest false rate and high score values of separation which are much of the most effective. Experiments conducted on word lists that are derived from the root word as group, set-based trigram and sumsquare gram methods have high values of similarity measure and performance to price with low running time of processing.

Figure 12: Sample of similarity testing using medical terms

Experimental results on subjective questions indicated that there was a moderate correlation between average expert score for each student and the similarity value of oddgram method. This showed that oddgram method seems to be the most promising by efficiently handling spelling and grammatical errors in subjective questions. The results achieved by the oddgram method was not exceptionally better than Dice similarity values but the running times with Dice method are highly encouraging and better than generalized n-gram matching. It was noted that the performance of set-based trigram in the similarity of programming languages was not satisfactory because most of the discriminating n-gram strings reside on unigram and bigram. It was also noted that the performance of the new five methods was not constrained to the number of the records of pattern and text matches due to the use of average and standard deviation

The evaluation methods are recall, precision, f-measure, k-nearest neighbour, Pearson correlation coefficient, highest false rate, lowest true rate, separation and stepwise grade to determine in what percentage degree of the proportional closeness to the expected results. The new methods that performed very well in their respective data sets, computed pattern and text matches with high values of similarity and performance to price with low rate of processing times which satisfy closeness measurement of similarity predicate.

(Adio Akinwale and Adam Niewiadomski, 2010, Adio Akinwale and Adam Niewiadomski, 2012, Adam Niewiadomski and Adio Akinwale, 2013, Adio Aknwale and Adam Niewiadomski, 2015).

2.1 Collaborative research works

A visualization concurrent control algorithms for real time database systems using model view controller was proposed together with visualization of database normalization algorithms.

The tools substantiate the earlier simulation-based performance studies by exposing spikes at same points when visualized dynamically that are not observed using usual static graphs (Folorunso and Akinwale, 2008).

Data mining and expert system were combined in designing an effective anomaly-based intrusion detection system which gave a better coverage and made the detection more effective. During the process, database was created which became the source of data for the usage in the intrusion detection system (Sodiya *et a,* 2004).

In collaboration with other scholars, data mining technique was applied to students' health database with a view to provide information about prevalent ailments and suggestions for managing the identified ailments. The results could be used by the management of the health centre for enhanced strategic decision making about health care (Onashoga *et al*, 2011).

A conceptual framework for an ontology-based examination system was designed to create meta data elements required for developing web-based application. The framework presented an application of the state of the art on ontology evaluation. The application showed that the choice of suitable method must depend on the theme of ontology evaluation (Adekoya F. A. *et al*, 2011).

A novel buying decision-making process, using an analytic hierarchy process (AHP) and a Multilayer Perceptron Neural Network (MLP-NN) with a library-publishers scenario as a case study was proposed to address the challenge of inconsistency in buyers' opinions. The AHP uses library books criteria and assigns a weight to each, using a pairwise comparison matrix with an arrangement based on priority. Consistency in judgement levels

was measured using congruence and dissonance. The MLP-NN was used to develop specific non-linear mapping by adjusting network weights using a learning algorithm. The weights were used to adjust coefficients using the least-squared error and gradient descent method to increase the consistency of buyers' opinions and consequently improve the choice of indexes that conformed to the common agreement of the buyers. The results show that the intensity of the process, in terms of number of alternatives and criteria, is dependent on both decision-making and knowledge acquisition (Vincent *et al*, 2017).

A secure mobile agent protocol was developed to manage network system. The system was secured with less bandwidth overhead for the purpose of efficiency. Dynamic protection architecture for mobile agent system using travel diary protection scheme and platform registry was suggested. The scheme protected and allowed mobile agents to roam freely in open network environment without being compromised in any malicious host (Ibharalu *et al*, 2011).

A new peer-assisted streaming model (AyoPeer) using game theoretic with the aim of minimizing free riding was presented. The interactions between peers are fashioned after a famous Yoruba game called "Ayo". The experimental results revealed that AyoPeer mechanism outperformed the existing scheme by ensuring fair distribution of resource, high cooperation among peers and resistance to free-riding peers (Oluwafolake E. Ojo *et al*, 2020).

Similarity measures have been extended into neutrosophic similarity measure for text retrieval. A new neutrosophic string similarity measure based on the longest common subsequence has been proposed to address uncertainty in string information search.

The new method has been compared with four existing classical string similarity measures using word list as data set. The analysis shows the performance of the proposed neutrosophic similarity measure to be better than the existing in information retrieval task as the evaluation is based on precision, recall, highest false match, lowest true match and separation (Alaran *et al* 2018).

Neutrosophic similarity measure has been proposed for the department selection for student transiting from JSS1 to SSS1 class in Nigerian education system. The result indicated a high percentage of accuracy of 82.86 for the proposed method from the Teachers who are mostly involved in this kind of exercise. An application also developed to enhance the usage of the new method (Alaran *et al*, 2019).

Similarity measures have been extended into intuitionistic fuzzy similarity measure with application in different domains. The research work modified existing fuzzy similarity measures and applied it to cognitive domain for better performance. Experiment showed that the proposed method gave higher similarity value and lower processing system (Hanat Y. Raji-Lawal *et al*, 2020).

2.2 Summary of completed, on-going and future research work

The proposed translated statements from untranslated statements using derived temporal method for database that attributes are in weak currencies. Translated statement was also proposed before performing database operations such as select, project, union, aggregation, cube, join, fetch (cursor), etc.

My research work also proposed five new similarity measures that compute similarity values of two matching patterns at the reduced processing time and high performance.

As a future work, effort would be made to use a combined

approach where multiple n-gram methods would be organized in a step-wise manner. Using similarity measure in case-based reasoning as a set of words in form of sentence, effort would be made to compare their stylistic parameter values such as average word length, average length of nouns, verbs and prepositions per sentence. By elaborating this idea of n-gram methods further, it is still believed that many new interesting methods could be developed and new interesting theoretical results would still be derived. The developed methods would be applied to other n-gram classification tasks. Any derived similarity measures would not be tied to a particular domain in order to determine where they are most fitted. The derived methods would be extended to other related fuzzy methods for relevant results.

2.3 Recommendations

There is a need to encourage the in-coming computer scientists to develop more interest in programming languages rather than engaging in hardware and system analysis. Computer Science Lecturers should be encouraged with modern Computer Science Laboratories to demonstrate more practicals in Computer Science Courses as being done in other courses like Physics, Chemistry and Biology. Government should encourage indigenous software applications in our Industries, Companies, Tertiary Institutions, etc, to promote home-made software applications in Nigeria. This will add values to the economy and employment. For unique Universities like that of the Federal University of Agriculture, there should be a unique Department of Agricultural Informatics that would combine expert's knowledge from both agricultural and science domains to bring innovative research benefits to the communities.

2.4 Acknowledgement

I am most grateful to the Almighty God who has always guided, protected and provided for me. I give Him all glory, honour and praise. I am very grateful to my wonderful parents late Mr Nosiru Akanbi Akinwale (died February 8, 1979) and late Mrs Rabiatu Abeke Akinwale (died March 29, 2015) who taught me hard work, diligence, respect, honesty, and ingrained in me the values of education early in life. They believed until their death that God is great and school is good. They did not mind going naked just to pay my school fees. Daddy and Mummy you were the best! I pray for forgiveness of their sins and illumination of their graves. I also thank all my siblings (Brother Lamina, Sisters Sikiratu, Awula, Sekinat, Kehinde and Bushiratu), my Uncles and Aunties for their love, encouragement and support in my journey of life in Nigeria and Poland.

My thanks are especially due to all my primary and secondary school teachers who made learning incredibly exciting. I acknowledge the contributions of Mrs Phillip, Mr Odutan, Akinrinade and Mr Olufunmilayo at the level of primary school who encouraged my parents to send me to the secondary school because of my performance. Otherwise, I could have been a painter. My great Principals who were disciplinarians in the various secondary schools I attended are highly appreciated for instilling discipline, laws and order in the schools which created a conducive atmosphere for us to pursue our studies. I mention particularly: late honourable George Olanipekun, Rev Okusanya, Mr Oluwasanmi, Mr Akinluyi, Mr Kobina, Mr EmBradry, Late Chief Femi Adewunmi, Mr Ajayi and Late Rev Oredein.

My both primary and secondary school classmates deserve special mention for their total cooperation during and after school periods.

They are Ikumapayi Abiodun, Fausat Adebayo, Sunday Oyekunle, Timothy Ogunseye, Tawa Olasunbo, Sunday Olufunmilayo, Funmilayo Sonde, Rotimi Adeleye, Prof Adegbite, Sesan Sobo, Dr Babatunde Ayorinde, Tajudeen Bada, Olawale Obey, Omobola Abbas and Monsuru Ayodeji. When I take a deep breath back at all the sweet memories we had shared together during our school days I cannot hold back my happiness.

There is a belief that friendship whose formation takes place at childhood is certainly the strongest. My special thanks go to Mr Dayo Omidiji, a retired principal who made it possible for me to attend the interview of External Bureau Scholarship Award by bringing the interview letter to my house. I thank you so much for your kindness, sincerity, and generosity. My heart just keeps thanking you and your family omission especially late Hon. Fola Omidiji for everything your family had done for mine. I am truly thankful to Elder Suraju Soneye for all of his kindness and care for me. I want to put it on record that you gave me my first pair of suits that I wore to Poland in October 1984. I cannot forget that act of kindness.

I am deeply honoured to be found worthy of receiving the External Bureau Scholarship Award by the Polish Government. I was at my wit's end when this scholarship opportunity came knocking on my door and took a heavy financial burden off my shoulder and it meant a lot to me and my family.

On my journey through life's path to this height, I acknowledge the contributions of some Lecturers at various points of my academic training, notably; Late Prof dr hab. Jacek Ochman, who supervised my first M.Sc dissertation, Prof dr hab. Jerzy Tyszkiewicz, supervisor of my second M.Sc dissertation, Prof inz dr hab Andrzej Baborski, who supervised my first Ph.D thesis and

Prof inz dr hab. Adam Niewiadomski, supervisor of my second Ph.D thesis. I need to thank them for their selflessness and I will never forget their contributions to my life. I would like to express my very great appreciation to my academic Lecturers: Prof dr hab Antonia Smoluk (linear algebra), dr Tadeusz Janaszak (mathematical analysis), Prof inz dr hab. Mieszyslaw Owoce (artificial intelligence) and Prof dr hab Andrzej Gospodarowicz (Pascal programming language).
My special gratitude goes to Prof dr hab. Krzysztof Diks (algorithm complexity and sequential), dr hab Marcin Engel (concurrent and imperative programming languages), dr Marcin Kubica (complier technique and computation theory), dr hab. Akeksy Schubert (networking programming and declarative programming), dr hab, Ewa Magdalinska (semantic programming language), dr Adams Malinnowki (discrete mathematics and data structure), Prof dr hab. Piotr Zakrzewski (abstract algebra), dr hab. Janina Mincer-Daszkiewicz (operating system and network operating system and mgr Zbigniew Jurkiewicz (database system and advanced database system). I have been so blessed by their contributions and willingness to help me fully grasp their materials. All the classroom scenes are ever fresh in my memory.

I would like to take this opportunity to thank Prof inz dr hab. Liliana Byszkowska-Lipinska for sponsoring my internship with the Institute of Informatics, Technical University, Lodz, Poland. I had an incredible experience. I was given opportunity to work in the computer laboratories with really intelligent people. I would like to thank Prof dr hab Mykhaylo Yatsymirskyy and dr inz hab Agnieszka Wosiak for taking time to discuss professional field of specialization with me. They demystified the field of my specialization and made me understand that computer science courses are around us and relatable.

My sincere thanks are especially due to Prof Adewale Roland Tunde Solarin who made my employment possible as a Lecturer I in this great University. I thank you for your leadership acumen and disposition when we were together in the defunct Mathematical Sciences Department.

Vice Chancellor Sir, kindly permit me to acknowledge my indebtedness to many people in this great University who had assisted in their respective capacities to facilitate this day by showering me their love. First among them is of course the 6th substantive Vice Chancellor, Prof Felix Kolawole Salako for giving me the opportunity to present this 68th Inaugural Lecture. I sincerely appreciate your soft skill that has added positive value to the peace and development of this unique Federal University of Agriculture, Abeokuta. I would like to express my deep gratitude to you Sir for your invaluable support in all aspects.

I would like to appreciate the following former Vice-Chancellors of the Federal University of Agriculture, Abeokuta for providing the enabling environment for me to work with them: under the 2nd Vice-Chancellor, Prof Julius Okojie, be approved my temporary appointment as Lecturer I and I worked as Co-ordinator of the defunct Mathematical Sciences Department; under the 3rd Vice-Chancellor, late Prof Isreal Adu, I was Overseer, Computer Science Department, under the Acting Vice-Chancellor, Prof Ishola Adamson, I was Acting Head, Computer Science Department, under the 4th Vice-Chancellor, Prof Olaiya Oluwafemi Balogun, he was the Chairman, Appointments and Promotions Committee for Academic Staff that considered and announced my Readership position; and Prof Bamidele Oyewole, the 5th Vice-Chancellor, who appointed me as Acting Dean, defunct College of Natural Sciences and also announced my Professorship position. Prof Olalade Enikuomehin deserves special mention here as he was the second Acting Vice-Chancellor

of this great Federal University of Agriculture, Abeokuta.

I would like to thank the present Deputy Vice Chancellor (Academic), Prof (Mrs) Bolanle Akeredolu-Ale, Deputy Vice Chancellor (Development), Prof O. B. Kehinde, the Registrar, Dr Adebola Adekola, the Bursar, Mr Chukwunwike Ezekpeazu and the Librarian. Dr (Mrs) Fehintola Onifade. My profound thanks go to the current Chairman, Committee of Deans and Directors, Prof W. O. Alegbeleye for his kind support in preparation for this Inaugural Lecture and his mutual cordiality to the College of Physical Sciences on Post-graduate matter. My thanks go to former Principal Officers, notably, Prof (Mrs) Morenike Dipeolu, Prof Lateef Sanni, Prof O. J. Ariyo, Prof C. F. I. Onwuka, Prof S. T. O. Lagoke, Prof T. A. Arowolo, Prof M. A. Waheed, Prof C. O. Adeofun and Prof I. C. Eromosele.

I am particularly grateful to my Head of Department, Prof O. Folorunso for his valuable and constructive advice during the planning of this Inaugural Lecture. My regards go to my colleagues with whom I have worked ceaselessly over the years and who have inspired me to provide the needed leadership. Let me specially mention Prof A. S. Sodiya, Prof O. A. Ojesanmi, Prof (Mrs) S, A, Onashoga, Dr (Mrs) O. T. Arogundade, Dr (Mrs) O. R. Vincent, Dr F. T. Ibharalu, Dr A. Abayomi-Alli, Dr D. O. Aborisade, Dr (Mrs) E. O. Ojo, Mrs C. O. Tinubu, Mrs A. O. Adejimi, Mr C. O. Ugwunna and Mrs Morenikeji Kareem.

My heart is full of gratitude to the other Heads of Department in the College of Physical Sciences, Prof (Mrs) Modupe Idowu, Prof B. I. Olajuwon, Prof B. S. Badmus and Dr O. M. Olayiwola. They are well organized and productive. I would like to appreciate former Deans of the College of Physical Sciences, Prof Oluyemisi Eromosele and Prof Amidu Mustapha for steering the ship and their continuous effort at re-branding the College so far. Their

repository of the knowledge and fate made me to follow them as the third Dean of the College. I thank also the present and past Deputy Deans, Prof. V. Makinde and Prof A. K. Akinlabi for their loyalty and support to run the College.

My fellow Professors in the College remain most appreciated for their mutual support and cordiality, notably; Prof J. A. Oguntuase, Prof O. J. Adeniran, Prof A. A. A. Agboola, Prof M. O. Omeike, Prof O. D. Akinyemi, Prof G. A. Adebayo, Prof (Mrs) I. C. Okeyode, Prof L. A. Arogundade, Prof E. A. Dare and Prof S. Adewuyi. My endless gratitude goes to up-coming Professors in the College, notably, Dr T. A. Afolabi, Dr A. I. Adeogun, Dr S. A. Amolegbe, Dr S. S. Sojinu, Dr (Mrs) T. F. Akinhanmi, Dr F. O. Oladoyinbo, Dr (Mrs) A. D. Adesina, Dr A. T. Aremu , Mr F. Akinwunmi, Dr S. A. Akinleye, Dr E. O. Adeleke, Dr I. A. Osinuga, Dr E. Ilojide, Dr M. T. Raji, Dr O. J. Ogunsola, Dr A. A. Adeyanju, Dr D. O. Adams, Mr A. A. Yusuff and Mr F. M. Nkwuda. Other notably up-coming Professors remain appreciated: Dr (Mrs) O. Alatise, Dr F. G. Akinboro, Dr J. O. Akinlami, Dr G. O. Layade, Dr A. A. Alabi, Dr O. T. Olurin, Dr S. A. Ganiyu, Dr O. P. Adebambo, Dr A. I. Egunjobi, Mr K. D. Ajayi, Dr S. O. N. Agwuegbo, Dr (Mrs) F. S. Apantaku, Dr G. A. Dawodu, Dr (Mrs) A. A. Akintunde, Dr (Mrs) O. A. Wale-Orojo, Dr A. T. Soyinka, Mr A. Ajayi and Mrs B. S. Adetona. They deserve special appreciations for their respective cares and concerns toward the running of the College.

I also appreciate the Dean of our Sister College, Prof O. A. Akinloye and the former Deans, the defunct College of Natural Sciences, Late Prof E. O Asiribo, Prof T. O. S. Popoola under whom I served as Deputy Dean and Prof A. D. Agboola. I would like to appreciate present Deans and former Deans of the various Colleges in this great University for the love to me and the

College of Physical Sciences notably, Prof M. O. Atayese, Prof O. S. Sowande, Prof I. T. Omoniyi, Prof E. O. Fakoya, Prof S. O. Ismail, Prof O. U. Dairo, Prof M. A. Idowu, Prof W. A. O. Afolabi, Prof (Mrs) B. B. Phillip, Prof A. O. Dipeolu, Prof C. O. Ikeobi, Prof J. G. Bodunde, Prof J. K. Adewunmi and Prof A. K. Akinloye. I appreciate all my research collaborators; Prof I. A. Adejumobi, Dr O. A. Akinola, Dr O. S. Abiola, Prof A. K. Akintokun, Prof Ayo-Vanghan, Prof O. A. Osipitan, Prof J. A. Olanite, Prof S. O. Oluwalana who always asks the welfare of my son, Prof A. M. Shittu, Prof G. C. Ufoegbune, Prof A. B. Idowu, Prof H. A. Bodunde, Prof O. Ademuyiwa and Prof B. A. Adewunmi.

I specially thank all my students at both undergraduate and postgraduate levels in Nigeria and Poland for the opportunity to learn from each other and engagement in research. My tribute must go to all non-academic staff that touched my life or whom I have directly worked with notably, Mr A. O. Agbotoba, Mrs O. M. Dawodu, Mrs A. O. Kuforiji, Mrs W. A. Badmus, Mrs O. O. Banuso and Mrs O. B. Adediran

The list of those to be acknowledged is indeed inexhaustible. I do profusely apologise to all those who deserve to be so acknowledged in this piece. Please forgive me because I am a gentle man of few words. God knows all of you and will not owe you anything.

To all those present here today including the umpteen around the world watching through webinar (zoom). I remain eternally indebted and hereby inform that comments and criticism remain most welcomed (akinwaleat@funaab.edu.ng) which will engender further research.

Mr Vice Chancellor Sir, we normally save the best for the last and my nuclear family occupies this eminent position. The members

of my family constitute the soul of this journey and the solid pillars of the structure. My children: Robert Akinwale, Krystian Akinwale, Sandra Akinwale, Victoria Akinwale and Victor Akinwale are my joy, fan base and pride. I deeply appreciate my amiable, excellent, diligent wife, Mrs Oluwatoyin Abosede Akinwale for the understanding and conducive home atmosphere that allowed me to reach this level of academic achievement. Your presence in my life has made everything beautiful. Thank you for being there as a pillar of support and encouragement and my prayer is that God will continually shield you, the children and I and bless us with good health and sound mind.

I thank you all for your kind attention.

God bless (amen).

References

Adam Niewiadomskiand **Adio Akinwale**, (2015), Efficient N-gram based String Matching in Electronic Testing at Programming, *Lecture notes in Computer Science*,Springer, 8083: 661-670

Adekoya Felix, **Akinwale Adio T.** and Sofoluwe Adetokunbe B, A (2011), Conceptual Framework for Ontology Based Examination System, *International Journal of Advanced Computer Science and Application*, 2(4):36-42

Adio Akinwale (1993), Designing accounting systems for processing financial interest of jointly owned companies, *Informatyka, Research Papers of Computer Science,* 660: 169-189

Adio Akinwale (1994), A computer-based decision support system for assessment of joint venture companies, *Informatyka, Research Papers of Computer Science,* 683: 23-35

Adio Akinwale (1994), Fragmentary knowledge base for translated statement operation, *Informatyka, Research Papers of Computer Science***,** 683: 11-19

Adio Akinwale (2001), Utilization of computer system in the banking queuing systems, *Informatyka, Research Papers of Computer Science,* 907: 115-120

Adio Akinwale, Olusegun Folorunso and Adesina Sodiya (2011), Derivation of database keys operations, *Journal of Issues of Informing and Information Technology*, 8(1):277- 290

Adio Akinwale and Adam Niewiadomski, (2012), An effective similarity measure in electronic tests at programming languages, *Journal of Applied Computer Science*, 20(2): 7-26

Adio Akinwale and Adam Niewiadomski, (2015), Efficient similarity measures for Texts Matching, *Journal of Applied Computer Science*, 23(1): 1-7

Adio Akinwale and Adam Niewiadomski (2010), New similarity measures in electronic tests at programming languages, *TEWI platforma informatczna*, 23: 1-14

A. T. Akinwale, O, J. Adeniran and O.T. Arogundade (2003), Exploiting semantic model for interactive query transformation using mix fix function, *ASSET series B,* 2(1):73-82

A. T. Akinwale and A. A. G. Akanmu (2005), Comparative study of the features oft web design tool, *ASSET series B*, 4(1): 75-81

A. T. Akinwale, O.T. Arogundade and H. Obianugba (2009), A way forward of improving functionalities of open database system, *Journal of Theoretical and Applied Information Technology*, 7(1): 001-007

Akinwale A.T. and Ibharalu F.T. (2009), Password authentication schemes with secured login interface, *Anale Seria Informatica*, 7(2): 77-85

Akinwale Adio T., O.T. Arogundade and A.F. Adekoya (2009), Translated Nigeria stock exchange price using artificial neural network for effective prediction, *Journal of Theoretical and Applied Information Technology*, 9(1):36-43

Akinwale Adio Taofiki, Adekoya Adebayo and Onashoga Adebukola (2010), An Egocentric database operations for social and economic network analysis, *Interdisciplinary Journal of Information Knowledge and Management*, 5(1): 293-303

Akinwale Adio Taofiki, Adekoya Felix Adebayo and Ooju Emmanuel (2011), Multi-level cryptographic function for the functionalities of open database system, *Journal of Computer Technology and Application,*2(9):730-735

Akinwale A.T., Adesina A. A. and Folorunso O. (2011), A framework for reducing multidimensional database to two-dimensional database, *Journal of Information and Computing Science*, 6(4):269-278

Akinwale Adio Taofiki (2011), Hierarchical database model for querying economic network independence distribution, *International Journal of Decision Support System Technology*, 3(3):58-67

Akinwale A. T., Adebayo A. A., Folorunso O. and Adebayo G. A., An Improved Selection Technique for Fast Packet Routing in Computer Network, *International Journal of Computing,* 10(3):223-234

Akinwale Adio Taofiki and Arogundade Tale Oluwasefunmi (2012), A visualization tool for teaching and learning database decomposition system, *Journal of Information and Computing Science,* 7: 003-010

Akinwale Adio Taofiki, (2012) A fast message packet forwarding using Fibonacci method, *Journal of Computer Sciences and*

Telecommunications, 3(35):64-69

Akinwale A. T., Shonubi A. J., Adekoya A. F., Sodiya A. S. amd Mewomo O. T. (2013), Ontology of input validation attack on web application, *Business informatics*, 4(30): 11-23

Akinwale Adio and Babatunde Daniel (2015), A packet forwarding using generating function of Fibonaaci method, *Journal of Computer Sciences and Telecommunications*, 3(35): 64-69

A. F. Adekoya and **A.T. Akinwale** (2003), Comparative analysis of the efficiencies of two optimization technique in distributed database system, *ASSET series B*, 2(1): 45-60

A.S. Sodiya, H.O.D. Longe and **A.T. Akinwale** (2004), A new two-tiered strategy to intrusion detection, *Information Management and Computer Security*, 12(1): 27-44

A. S. Sodiya, H.O.D. Longe and **A. T. Akinwale** (2005), Maintaining privacy in anomaly-based intrusion detection systems, *Information Management and Computer Security*, 13(1):72-80

Folorunso O. and **A. T. Akinwale** (2008), A conceptual analysis and management of Ifa (An African traditional oracle), *Kybernetes, International Journal of Cybernetics, Systems and Management Sciences*, 38(4): 625-634

F.T. Ibharalu and **A. T. Akinwale** (2003), Multi-level hash tables with multi-hash function, *ASSET series B,* 2(1): 127-140

Hanat Y. Raji-Lawal, **Adio T. Akinwale,** Olusegun Folorunsho

and Amidu O. Mustapha, (2020), Decision support system for dementia patients using intuitionistic fuzzy similarity measure, *Journal of Soft Computing Letters*, Elsevier 2:11-19

Harrington, J., and Harrington, J. (2016). *Relational database design and implementation : Clearly explained* (4th ed.). Amsterdam: Morgan Kaufmann/Elsevier.

Ibharalu Thomas Friday, Sofoluwe Adetokunbo Babatunde and **Akinwale Adio Taofiki**, (2011), A Reliable Protection Architecture for Mobile Agents in Open Network System, *International Journal of Computing and Application* 17(7):6-14

Misturah Adunni Alaran, AbdulAkeem Adesina Agboola, **Adio Taofiki Akinwale** and Olusegun Folorunso (2018), A new LCS-Neutrosophic similarity measure for text information retrieval, *Journal of intelligence and fuzzy systems*, 12:258-279

Misturah Adunni Alaran, AbdulAkeem Adesina Agboola, **Adio Taofiki Akinwale** and Olusegun Folorunso (2019), A neutrosophic similarity approach to selection of department for student transiting from JSS3 to SSS1 class in Nigerian education system, *Journal of neutrosophic sets and system*, 27:104-113

O.T. Arogundade and **A.T. Akinwale** (2007), The use of conceptual query processing in livestock management system, *Asian Journal of Information Technology*, 6(1):170-175

O. Folorsunso, Longe H.O. D. and **Akinwale A.T.** (2008), Visualizing concurrency control algorithm for real time database system, *Data Science Journal and Technology Aggregator Electronic*, 7(1): 96-105

Oluwasefunmi Arogundade, **Adio Akinwale**, Zhi Jim, and Xiaoguang Yung, (2011), Venerability analysis approach to capturing information system threats and requirements, *International Journal of Software Engineering and its Application*, 5(4): 85-98

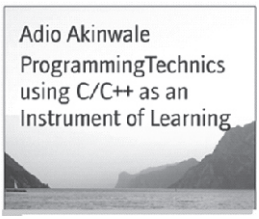
Olufunke Rebecca Vincent, Ayodeji Samuel Makinde and **Adio Taofiki Akinwale**, (2017), A cognitive buying decision-making process in B2B e-commerce using analytic-MLP, *Journal of electronic research and application*, 25: 50-69

Oluwafolake E. Ojo, Cyril O. Ijadi, Ayodeji O. Oluwatope and **Adio T. Akinwale,** (2020) AyoPeer: The adapted ayo-game for minimizing free riding in peer-assisted network, *Journal of peer-to-peer networking application*, Springer, 13(4):1672-1787

Onashoga S.A., Sodiya A.S., **Akinwale A.T.** and Falola O.E. (2011), Mining of Prevalent Ailment in a Health Database using FP-Growth Algorithm, *International Journal of Data Engineering,*2(2):75-83

Onashoga S. A., **Akinwale A. T.**, Amusa O. and Adebayo G. A. (2019), CKRAID: Collaborative knowledge repository for intrusion detection system, *Journal of Computing and Information Technology,* 27(2): 29-39

Adio Akinwale, (2020), Programming technics using C/C++ as an instrument of learning, *BookRix*www.bookrix.com



University Senate Building

ISBN: 978-978-991-965-9