Adama Science and Technology University

College of Engineering

Department of Computing

Software engineering Program (Post Graduate)

Cloud Based Plagiarism Detection SaaS for Higher Educations and Institutes of Ethiopia

BY:

LIJALEM MEGIBARU

Advisor: Dr. Dereje Yohannes (Ph.D.)

June 21, 2016

Adama, Ethiopia

Adama Science and Technology University

College of Engineering

Department of Computing

Software engineering Program (Post Graduate)

Cloud Based Plagiarism Detection SaaS for Higher Educations and Institutes of Ethiopia

A Thesis Submitted in Partial Fulfillment of the

Requirement for the Degree of

Master of Science in Software engineering

BY:

LIJALEM MEGIBARU

June 21, 2016

Adama Science and Technology University

College of Engineering

Department of Computing

Software engineering Program (Post Graduate)

Cloud Based Plagiarism Detection SaaS for Higher Educations and Institutes of Ethiopia

By:

Lijalem Megibaru

Name and signature of members of the examining board,

Name	Title	Signature	Date
1. Mr. Mohammed Kemal	(Chair Person)	_____	6/21/2016
2. Dr. Dereje Yohannes (PhD)	(Advisor)	_____	6/21/2016
3. Dr. Tibebe Beshash (PhD)	(External Examiner)	_____	6/21/2016
4. Dr. Frezewd Lemma (PhD)	(Internal Examiner)	_____	6/21/2016

Declaration

The thesis is my original work, has not been presented for a degree in any other university and all sources of materials used for the thesis have been acknowledged.

_____

Lijalem Megibaru

This thesis has been submitted for examination with my approval as university advisor.

_____

Dr. Dereje Yohannes (Ph.D.)

June 21, 2016

Acknowledgement

I feel a great sense of accomplishment to be able to complete this thesis, as part of the requirement for My Master's degree. I would have not been able to do this on my own without the supports of the people whom are important to me next to the **Almighty GOD.**

Firstly, I would like to thank my supervisor, **Dr. Dereje Yohannes (Ph.D.)** for his continuous support and guidance. His advice and comments are crucial in my progress to complete this thesis.

I would also like to thank my family and parents for their words of encouragement and their supports throughout my study period. My friends are also important to me, for without them this experience would be less meaningful.

Abstract

Now a days, plagiarism detection has come a burning issue that academics have been facing in research areas especially in higher educations. With the collaboration of cloud computing and the plagiarism detection methods, the challenge faced higher educations of Ethiopia will be simplified. This paper introduces plagiarism detection approach which is designed based on cloud computing technology to be implemented in higher educations of Ethiopia. The main feature of these research is designing and implementing the plagiarism detection system, which is developed by combining the cloud computing future and plagiarism detection methodologies and algorithms. The system uses standard web search engines to locate documents on the cloud that might have been used as sources of plagiarism by the researcher.

Key Words: Plagiarism Detection, Cloud computing, plagiarism checker SaaS,

Table of Contents

List of tables

List of Figures

List of Acronyms and Abbreviations

GUI	Graphical User Interface
LAN	Local Area Network
DBMS	Database Management System
AWS	Amazon Web service
SAAS	Software as a Service
PAAS	Platform as a Service
IAAS	Infrastructure as a Service
IDE	Integrated development environment
API	Application programming Interface
VPN	Virtual private network
SLA	Service level agreement
VM	Virtual machine
ORM	Object Relational Mapping
MVC	Model View Controller
LAMP	Linux Apache MySQL PHP
MOE	Ministry Of Education

Chapter One

1. Introduction

Plagiarism is [1] a very common phenomenon now days and it is center of discussion at various educational events. Plagiarism is defined as the practice of claiming or implying original authorship of (or incorporating material from) someone else's written or creative work, in whole or in part, into one's own without adequate acknowledgement. E-learning programs has touched every corner of this small world due to the advanced in the Information Technology , this in turn led to easier availability of the research papers , books , technical and non-technical papers which are easiest source for making plagiarized documents [1].

There are many types of plagiarism, these are copy and paste, redrafting or paraphrasing of the text, plagiarism of idea, and plagiarism through translation from one language to another. These types have made plagiarism one of the serious problems in academic area.

Commonly there are different plagiarism methods practically applied. Some of these methods are include [2]:

- Copy – paste plagiarism (copying word to word textual information)
- paraphrasing (restating same content in different words)
- translated plagiarism (content translation and use without reference to original work)
- Artistic plagiarism (presenting same work using different media: text, images etc.)
- Idea plagiarism (using similar ideas which are not common knowledge)
- Code plagiarism (using program codes without permission or reference)
- No proper use of quotation marks (failing to identify exact parts of borrowed content)
- Misinformation of references (adding reference to incorrect or non-existing source)

There are several approaches to plagiarism detection that have evolved. The diagram describes the approaches used during plagiarism detection.

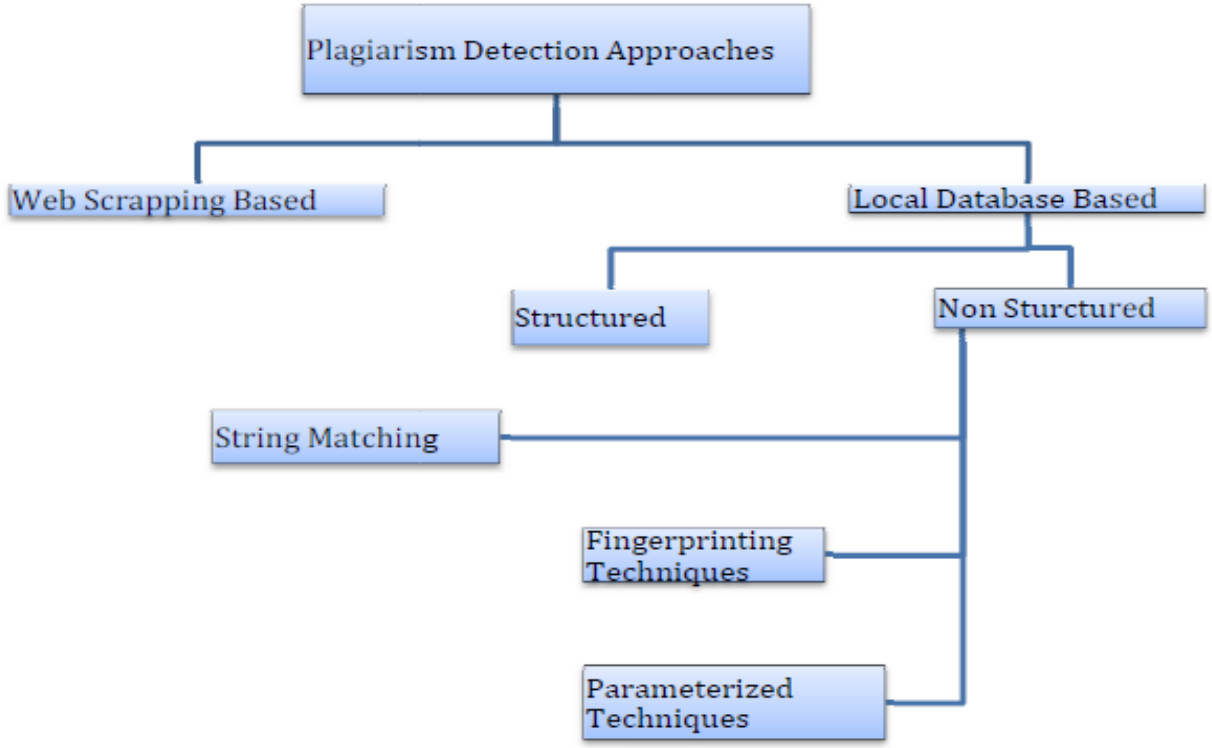

Figure 1: Approaches to plagiarism detection *[3]*

Plagiarism is on the rise. Manual detection of plagiarism is a very tedious and often unproductive task. Manual plagiarism detection is not always feasible as large amount of expert time required for the analysis. It may not be possible for one person to detect plagiarism in a document, because it requires that the person is knowledgeable in almost all the sources from which parts of the content may have been taken and inserted into the plagiarized document. Hence, it generally takes a group of knowledgeable people to classify a document as plagiarized or not, which means that a lot of expert time is spent on the detection. Hence, in order to have plagiarism in check, we need automated plagiarism detection mechanisms.

Now a days there are various tools for plagiarism detection for example Turnitin [3], CHECK [4], SCAM [5], and MOSS [6], which are capable of detecting a few types of plagiarism like word for word plagiarism, paraphrasing and code plagiarism. In these research since our focus is based on the document or text plagiarism, the code plagiarism will not be discussed here.

Text based plagiarism detection process stages are categorized as the following [7]

1) Stage One Collection: This is the first stage of Plagiarism Detection Process, and it entails the student or researcher to upload their assignments or works to the web engine, the web engine acts as an interface between the students and the system.

2) Stage Two Analysis: In this stage all the submitted corpus or assignments are run through a similarity engine to determine which documents that are similar to other documents. There are two types of similarity engines, first intracorpal engine and second extracorpal engine. The intracorpal engines work by returning ordered list between each similar pairs. By contrast, the extracorpal engines return suitable web links.

3) Stage Three Confirmations: The function of this stage is to determine if the relevant text has been plagiarized from other texts or to determine if there is a high degree of similarity between a source document and any other document.

4) Stage Four Investigation: This is the final stage of a Plagiarism Detection Process and it relies on human intervention. In this step a human expert is responsible for determine if the system ran correctly as well as determining if a result has been truly plagiarized or simply cited.

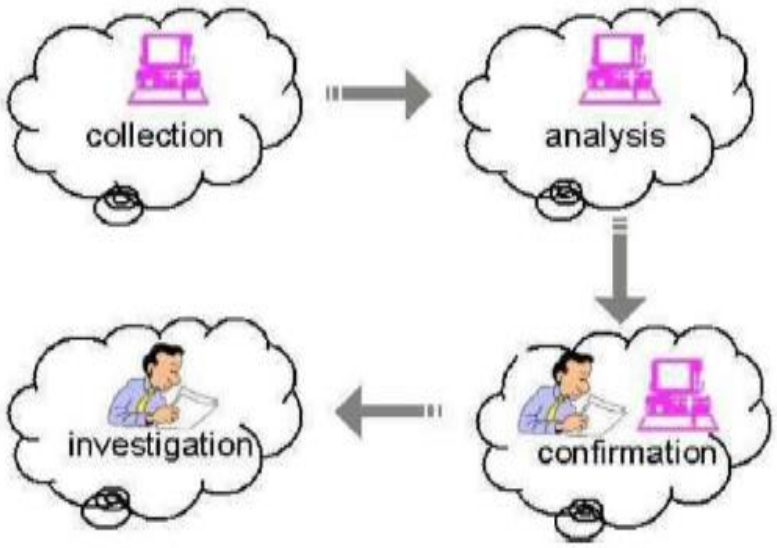

Figure 2: Stages of Plagiarism Detection Process *[1]*

In general this study will select the best plagiarism detection approach that helps higher educations in Ethiopia to control dissemination of research resources and the solution cloud based three tier architecture will be applied.

1. 2. Statement of the Problem

In some of the academic enterprises like universities, schools and institutions, plagiarism detection and prevention became one of the educational challenges, because most of the students or researchers are cheating when they do the assigned researches and projects. This is because a lot of resources can be found on the internet. It is so easy for them to use one of the search engines to search for any topic and to cheat from it without citing the owner of the document. So it is better and must all academic fields they should have to use plagiarism detection mechanisms to stop or to eliminate students cheating, copying and modifying documents. Plagiarism detection is a very important issue today because in the era of the Internet and digital media, resources are available and accessed. Since higher educations have no any mechanism to check the findings of researchers in other institutes, they simply consider the researches are the findings of the host researcher, and

that's why this study recommend the power of the cloud technology to bring plagiarism detection techniques to higher educations of Ethiopia.

So in order to overcome the dissemination of plagiarism in Ethiopia, implementing these plagiarism detection system which is developed based on string matching algorithm with the n gram approach is proposed.

1.3. Objective of the study

1.3.1. General objective

The general objective of this study is to propose cloud based software as a service system for detecting plagiarism in higher educations of Ethiopia based on selected plagiarism detection techniques and algorithms.

1.3.2. Specific objectives

The specific objectives of the study are:

- Implementing cloud computing technologies to control plagiarism detection.
- Selecting the plagiarism detection technique that will be effective in case of higher educations of Ethiopia.
- Propose the best plagiarism detection algorithm to be implemented in these system
- Suggesting the type of deployment model used in the cloud to detect plagiarism.
- Designing new proposed architecture that demonstrates the implementation of cloud technology in plagiarism detection task.

1.4. Approach and Methods

Extensive review of acknowledged texts, standards documents, industry periodicals and white papers, analysts' reports and conference journals is done in order to have detail understanding on

this research work. By applying cloud technology futures, all higher educations and institutions in Ethiopia will form a community cloud and they all are connected through the cloud using virtualization technology. The Plagiarism detection method of this paper is by comparing the document against a body of documents which is string matching algorithm. This approach can be further divided into two categories; one that operates locally on the client computer and does analysis on local databases of documents or performs internet searches, the other is server based technology where the user uploads the document and the detection processes take place remotely. The result gives a similarity approximation among documents being checked. Literature review is my main methodology. After reviewing the papers or researches done for plagiarism detection, we select the better plagiarism detection techniques, approaches and algorithms in the Ethiopian higher educations context. This sustainable plagiarism detection service for higher educations would need to satisfy the following requirements:

- Provide a seamless platform for servicing students, educators, and administrator
- Provide offsite management and administration to remove liability and simplify processes for educational institutions
- Provide plagiarism detection and prevention services through the software as a service (SaaS) model (software on demand) to simplify billing, administration and use of supported services
- The service must be accessible from any location to virtualized access and limit physical hardware requirements
- Support a variety of devices and terminals for accessing the plagiarism detection and prevention service.

1.5. Significance of the Study

This study has a great contribution for higher education institutes in Ethiopia.

- Encourage creativity among researchers
- Recognizes and motivates researchers
- Increase the collaboration between universities and researchers
- Used for sharing resources (Large volume of Literature will be available on single Click.)
- Less Violation of Copy Rights (for researchers)
- Intellectual Interaction through cloud, the papers will be stored on the cloud database and then findings will help researchers to find another gap.
- Security
- Centralization is the core issue

Generally, from this system there are many stakeholders which will be beneficiary, for example, the researchers, the institutes or higher educations and the users that needs to access the publications will be those who will be the parts of the system.

1.6. Scope and Limitation of the study

The main aim of these research is introducing the best quality of using fast and effective plagiarism detection methods and approaches in combine with cloud technology to detect plagiarism which is a common problem in higher educations particularly in Ethiopia context. Here the study is merely concentrated on designing simple plagiarism detection system using a cloud technology concept by implementing the already existing plagiarism detection techniques and algorithms.

So the scope of the study is limited with designing the plagiarism detection system for higher educations of Ethiopia based on the cloud computing concept.

When we come to the limitation of the research, there are some issues which will not be addressed in these thesis. Since these plagiarism detection system is mainly focused in Ethiopian higher educations and institutes context, for the time the research is not concerned about the global issue. The second limitation of the plagiarism detection system its only focus is on the text based detection, rather the source code, audio and video formats of the plagiarism.

1.7. Research Questions

This study is intended to answer the following research questions

- What is the contribution of the research for the community?
- What cloud deployment model is preferred to be used to detect plagiarism?
- What plagiarism detection technique is used in this system?
- What plagiarism detection algorithms are used to enhance the plagiarism detection system?

1.8. Thesis structure

Chapter one describes the problem setting, the research motivation, and the corresponding research objective. The research objective is divided into multiple research tasks pursued in this thesis.

Chapter two introduces the reader to the detail of cloud computing technology. Following a definition cloud computing and the deployment models, the types of cloud technology, cloud computing components and related technologies are discussed. This chapter addresses the general overview of cloud computing technology. As well it provides literature review on plagiarism detection mechanisms. The most relevant and similar papers are reviewed and discussed in detail. As well as the different algorithms used in plagiarism detection tools are discussed.

Chapter three presents the methodology the research follows and the approaches proposed in this thesis.

Chapter four describes the proposed plagiarism detection system design and analysis using cloud technology. Here we tried to explain deeply about the concept behind plagiarism detection techniques and the algorithms used in plagiarism detection process. And we discuss the recommended type of plagiarism detection approach which is text comparison and finally the proposed plagiarism detection algorithm which is used to enhance the plagiarism detection system with cloud technology to detect plagiarism is presented. And the plagiarism detection architecture used in this system also discussed.

Chapter five provides a discussion and chapter six discusses about the summary, research contributions, and gives an outlook on future work.

Chapter Two

2. Background and Related works

2.1. Cloud computing concepts

In recent years, cloud computing as a new kind of advanced technology accelerates the innovation for the computer industry. Cloud computing is a computing model based on networks, especially based on the Internet, whose task is to ensure that users can simply use the computing resources on demand and pay money according to their usage by a metering pattern similar to water and electricity consumption. [8]

Therefore, it brings a new business model, where the services it provides are becoming computing resources. Cloud computing is highly scalable and creates virtualized resources that can be made available to users. Users do not require any special knowledge about the concept of Cloud computing to connect their computers to the server where applications have been installed and use them. Users can communicate through Internet with remote servers. These servers can exchange their computing slots themselves. Cloud computing is one of the new technology trends likely to have a significant impact on the teaching and learning environment. In Cloud computing, resources can be either externally owned (public Cloud – as provided by Google and Amazon) or internally owned (private Cloud). Public Clouds offer access to external users who are typically billed on a pay-as you-use basis. The private Cloud is built for the access within the enterprise where the users can utilize the facility without any charge. The methods of meeting challenges such as user interface; task distribution and coordination are explained and evaluated in. They have discussed architecture as well as applications. Cloud computing attributes can be visualized from the following comparison.

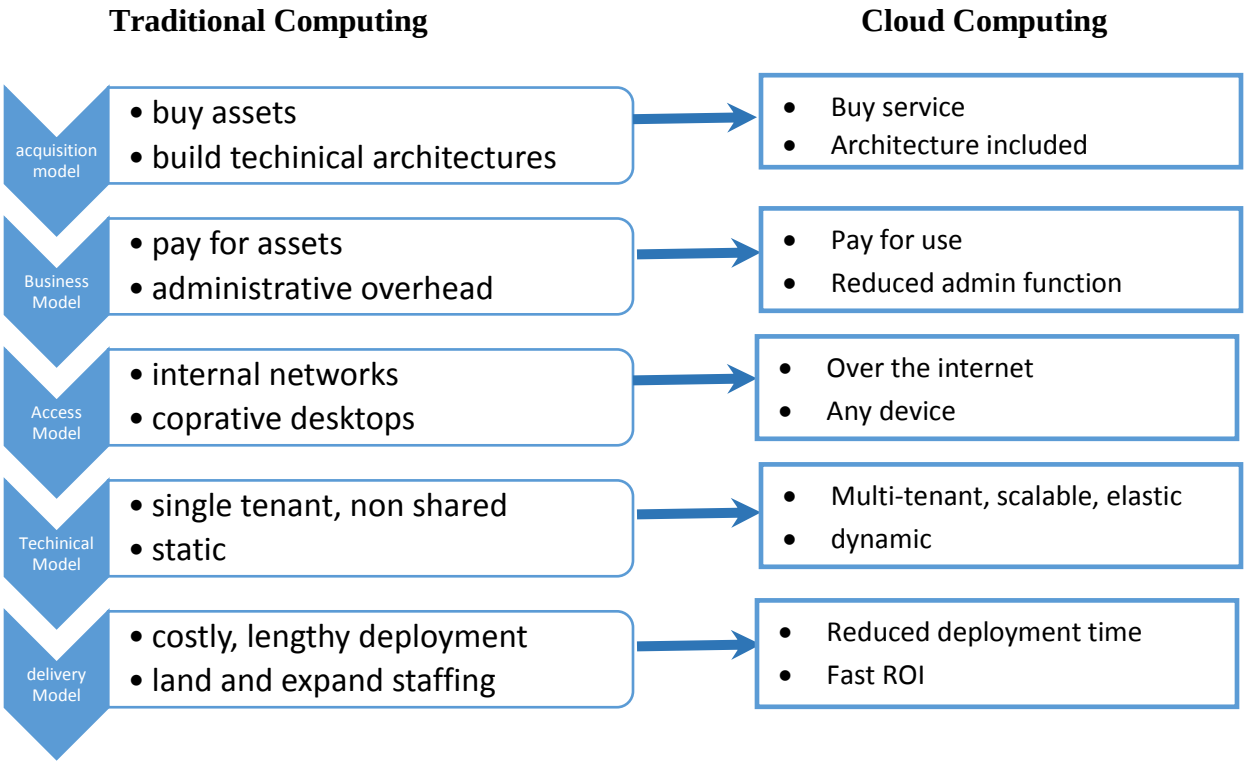

Figure 3: comparison of traditional computing and cloud *[8]*

[9] [10]**On-demand self-service**: A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service's provider.

Broad network access: Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, laptops, and PDAs).

Resource pooling: The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources

but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter). Examples of resources include storage, processing, memory, network bandwidth, and virtual machines. [11]

Rapid elasticity: Capabilities can be rapidly and elastically provisioned, in some cases automatically, to quickly scale out and rapidly released to quickly scale in. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be purchased in any quantity at any time.

Measured Service: Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported providing transparency for both the provider and consumer of the utilized service.

More and more information on individuals and companies is placed in the cloud; concerns are beginning to grow about just how safe an environment it is? Issues of cloud computing [9] can summarize as follows:

A. Privacy : Cloud computing utilizes the virtual computing technology, users' personal data may be scattered in various virtual data centers rather than stay in the same physical location, users may leak hidden information when they are accessed cloud computing services. Attackers can analyze the critical task depend on the computing task submitted by the users.

B. Reliability: The cloud servers also experience downtimes and slowdowns as our local server.

C. Legal Issues: Worries stick with safety measures and confidentiality of individual all the way through legislative levels.

D. Compliance : Numerous regulations pertain to the storage and use of data requires regular reporting and audit trails. In addition to the requirements to which customers are subject, the data centers maintained by cloud providers may also be subject to compliance requirements.

E. Freedom: Cloud computing does not allow users to physically possess the storage of the data, leaving the data storage and control in the hands of cloud providers. [12]

F. Long- Term Viability : You should be sure that the data you put into the cloud will never become invalid even your cloud computing provider go broke or get acquired and swallowed up by a larger company.

2.2. Cloud computing building blocks

A. Deployment models

Table 1. Comparative study between public, private and hybrid cloud *[13]*

Public cloud	**Private cloud**	**Hybrid cloud**
Availability of Elastic and Flexible Environment	Scope of Security and Confidentiality	Offer flexibility, control and security
Pay for Use Service	Greater Customization:	Data Centre consolidation
Freedom of self service	Maximum Protection	Risk transfer of workload
Resource Availability and Reliability	Own Dedicated Resources	Optimum Utilization of resources
Amazon EC2, Google App Engine, IBM Blue Cloud and Widows Azure	Amazon Virtual Private cloud, Eucalyptus Cloud Platform, IBM Smart Cloud, Microsoft Private Cloud	Requirement of both on-premises resources and remote server based cloud infrastructure

In the cloud deployment model, networking, platform, storage, and software infrastructure are provided as services that scale up or down depending on the demand. The Cloud Computing model has four main deployment models which are: [14]

1. Private Cloud

Private cloud is a new term that some vendors have recently used to describe offerings that emulate cloud computing on private networks. It is set up within an organization's internal enterprise datacenter.

In the private cloud, scalable resources and virtual applications provided by the cloud vendor are pooled together and available for cloud users to share and use. It differs from the public cloud in that all the cloud resources and applications are managed by the organization itself, similar to Intranet functionality. Utilization on the private cloud can be much more secure than that of the public cloud because of its specified internal exposure. Only the organization and designated stakeholders may have access to operate on a specific Private cloud. One of the best examples of a private cloud is Eucalyptus Systems.

2. Public Cloud

Public cloud describes cloud computing in the traditional mainstream sense, whereby resources are dynamically provisioned on a fine-grained, self-service basis over the Internet, via web applications/web services, from an off-site third-party provider who shares resources and bills on a fine-grained utility computing basis. It is typically based on a pay-per-use model, similar to a prepaid electricity metering system which is flexible enough to cater for spikes in demand for cloud optimization. Public clouds are less secure than the other cloud models because it places an additional burden of ensuring all applications and data accessed on the public cloud are not

subjected to malicious attacks. Examples of a public cloud include Microsoft Azure, Google App Engine.

3. Hybrid Cloud

Hybrid cloud is a private cloud linked to one or more external cloud services, centrally managed, provisioned as a single unit, and circumscribed by a secure network. It provides virtual IT solutions through a mix of both public and private clouds. Hybrid Cloud provides more secure control of the data and applications and allows various parties to access information over the Internet. It also has an open architecture that allows interfaces with other management systems. Hybrid cloud can describe configuration combining a local device,

Such as a Plug computer with cloud services. It can also describe configurations combining virtual and physical, collocated assets -for example, a mostly virtualized environment that requires physical servers, routers, or other hardware such as a network appliance acting as a firewall or spam filter. An example of a Hybrid Cloud includes Amazon Web Services (AWS).

Features of Hybrid Cloud [15]

Cost Effective- The demand nature of service allow company to rescue money, while also saving from IT staff deficiency since the service is to the full extend in order by the provider. Company can provide infrequent services using extensively computing resource from cloud service provider, and they quickly attach and recapture IT efficiency to meet peak and fluctuating service demand while only paying for actual capacity used.

Increase Storage - Provider can supply reasonable amount of storage space than an individual company may have or buy on its own.

Flexibility - Cloud computing offers efficient flexibility and adoptive approach to enterprise and corporate challenging environment.

Mobility - Global access is one of the key benefits of cloud computing i.e. any user can access any desired service from anywhere.

Scalability - Provision for rapid or quick and flexible service deployment, according to unpredictable and ever changing service, requirement, demand, enterprise and corporate environment.

Energy Efficiency - Generally cloud systems requires less energy compared to large data center, server form has 24/7 energy requirement. And also require power to cool down the server and data center as they release huge amount of heat energy. Whereas cloud system requires low maintenance and thus saves power.

Eco-Friendly - These traditional work station such as large data center and server farm have high Carbon footprints as these have huge energy consumption, therefore it has negative effect on environment.

Data Security - As most of the data of devices such as laptop, tablet, and mobile are saved or stored on cloud storage, it will remain safe even though devices may get stolen, damaged, misplaced and corrupted, you can still access your personal files from anywhere through your account. [15]

4. Community Cloud

Infrastructure shared by several organizations for a shared cause and may be managed by them or a third party service provider and rarely offered cloud model. These clouds are normally based on an agreement between related business organizations such as banking or educational

organizations. A cloud environment operating according to this model may exist locally or remotely. An example of a Community Cloud includes Facebook.

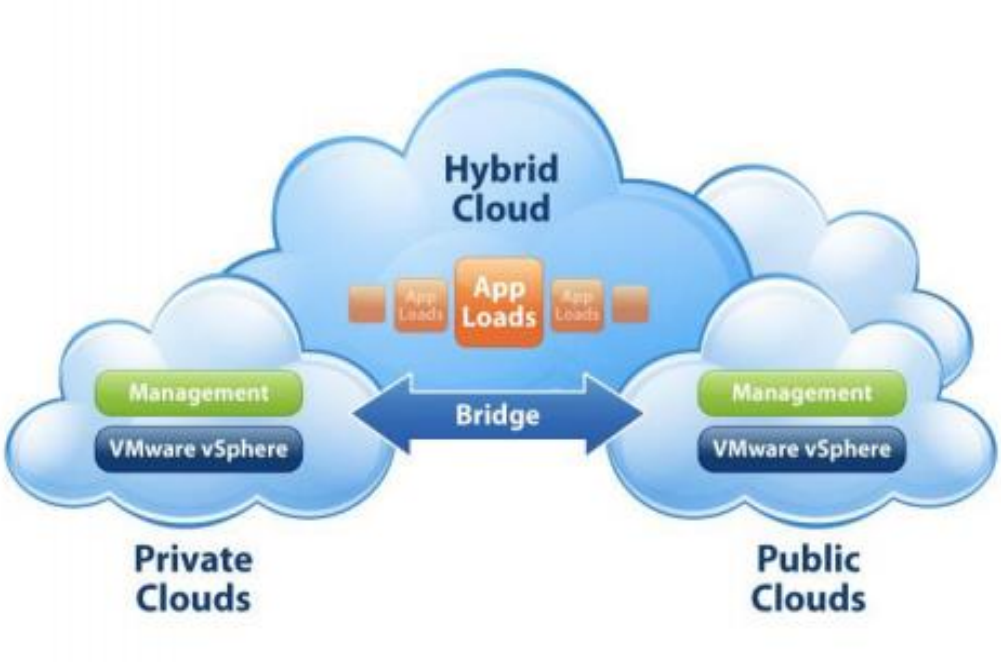

Figure 4: Cloud deployment models *[14]*

B. Service models

[16] According to the different types of services offered, cloud computing can be considered to consist of three layers: software as a service (SAAS), platform as a Service (PAAS), and infrastructure as a Service (IAAS). Infrastructure as a Service (IaaS) is the lowest layer that provides basic infrastructure support service. Platform as a Service (PaaS) layer is the middle layer, which offers platform oriented services, besides providing the environment for hosting user's applications. Software as a Service (SaaS) is the topmost layer which features a complete application offered as service on demand.

1. Software-as-a-Service (SaaS)

SaaS can be described as a process by which Application Service Provider (ASP) provide different software applications over the Internet. This makes the customer to get rid of installing and operating the application on own computer and also eliminates the tremendous load of software maintenance; continuing operation, safeguarding and support. SaaS vendor advertently takes responsibility for deploying and managing the IT infrastructure (servers, operating system software, databases, data center space, network access, power and cooling, etc.) And processes (infrastructure patches/upgrades, application patches/upgrades, backups, etc.) required to run and manage the full solution. SaaS features a complete application offered as a service on demand. In SaaS, there is the Divided Cloud and Convergence coherence mechanism whereby every data item has either the ―Read Lock or ―Write Lock. Two types of servers are used by SaaS: the Main Consistence Server (MCS) and Domain Consistence Server (DCS). Cache coherence is achieved by the cooperation between MCS and DCS. In SaaS, if the MCS is damaged, or compromised, the control over the cloud environment is lost. Hence securing the MCS is of great importance. Examples of SaaS includes: Salesforce.com, Google Apps.

[15]It generally means providing software or application over internet (generally termed as cloud in case of cloud computing) is termed as SaaS. It may also referred to as -On Demand Software. Thus, it can be provided according to the client requirement, demand, request and necessity and generally follows pay per use approach or scheme for payment.

2. Platform as a Service (PaaS)

PaaS is the delivery of a computing platform and solution stack as a service without software downloads or installation for developers, IT managers or end-users. It provides an

infrastructure with a high level of integration in order to implement and test cloud applications. The user does not manage the infrastructure (including network, servers, operating systems and storage), but he controls deployed applications and, possibly, their configurations. Examples of PaaS includes: Force.com, Google App Engine and Microsoft Azure.

[15]When platform such as Computing platform which generally includes Operating System, Integrated Development Environment (IDE) For Programming Language Execution, Database and Web Server. Any developer can develop their software and application and deploy it over cloud worrying about underlying hardware and software requirement as cloud automatically scales or adopt itself to changing environment and necessity. Computing service over cloud is known as PaaS (Platform as a Service). E.g. Microsoft Azure and Google App Engine.

3. Infrastructure as a Service (IaaS)

Infrastructure as a service (IaaS) refers to the sharing of hardware resources for executing services using Virtualization technology. Its main objective is to make resources such as servers, network and storage more readily accessible by applications and operating systems. Thus, it offers basic infrastructure on -demand services and using Application Programming Interface (API) for interactions with hosts, switches, and routers, and the capability of adding new equipment in a simple and transparent manner. In general, the user does not manage the underlying hardware in the cloud infrastructure, but he controls the operating systems, storage and deployed applications. The service provider owns the equipment and is responsible for housing, running and maintaining it. The client typically pays on a per-use basis. Examples of IaaS include Amazon Elastic Cloud Computing (EC2), Amazon S3, and GoGrid.

[15]When Infrastructure such as Virtualization feature is offered over or through internet then it is known as Infrastructure as a service or IaaS. It may include Virtual Private Network (VPN), providing Operating System through virtual machine over internet.

2.3. Cloud components

Cloud Computing does not refer to a specific technology, but a combination of pre-existing technologies and protocols that made this paradigm possible. The most relevant technological and non-technological components which support Cloud Computing architectures.

2.3.1. Technological components

Data centers and server farms

Cloud services require large amounts of computing capacity in order to provide high flexibility and power, and are hosted in data centers and server farms (which can be distributed in multiple locations).

For example, Google links a large number of servers to provide their services, or Amazon EC2 provides virtualization in a data center to create a huge amount of virtual instances.

Virtualization technologies

The most important technology that makes cloud computing possible. Virtualization technologies abstracts physical computing resources (CPU, storage, network, memory, databases...) from applications and end users, providing the capability of creating pooled resources accessible to anything authorized to use them.

This capability provides the multi-tenancy properties of cloud-computing, providing scalable and shared resource platform capabilities for all tenants. Virtualizations technologies are supported by

the hypervisor, a layer of software between an operating system and hardware platform that is used to operate Virtual Machines like launch or terminating, and implements and manages virtual CPU (vCPU), virtual memory (vMemory), event channels and memory shared by the hosted Virtual Machines (VMs), controls data, Input/output and memory access to devices.

When talking about virtualization applications, common users think on applications to create guest OS Virtual Machines like VMWare or VirtualBox, creating instances of Operating Systems that share resources with the host physical machine and other Virtual Machines. However virtualization does not limit to operating systems, but other components that can be virtualized, like storage (Storage Area Networks), databases, and software (Apache Tomcat, WebSphere, Jboss...). Depending on the delivery model, virtualization appears at various layers. In IaaS commonly storage and OS is virtualized, while on SaaS and PaaS databases and software are the typically virtualized component.

Application Programming Interfaces (APIs)

APIs provide features to customers, like self-provisioning and control of services and resources, and allow communication between foreign applications and the cloud service. The type of APIs depend on the delivery model, and can go from simple URL manipulations to SOA (Service Oriented Architecture) programming models.

However each CSP usually has its unique API and no standards exist on this matter, which makes cloud applications not portable among clouds, harming interoperability and locking-in the customer into the vendor's cloud. Also, IT staff is required to become familiar with those platform-specific features (although some organizations like the Universal Cloud Interface, which attempts to unite APIs from different cloud providers to create and open and standardized cloud interface).

Encryption

Encryption is the process of encoding messages or information in a way that that only authorized parties can read that information, preventing unwanted parties and hackers read it. This is usually done with the use of an encryption key, which specifies how the message will be encoded. Then, the authorized party uses a secret decryption key to decipher the message and read it. Encryption is the common technique used to protect the confidentiality of messages.

2.3.2. Non-technological components

Service-level Agreements (SLA)

The mutual contract between providers and users is usually called SLAs which offers guarantees on the quality of the service by defining what is to be expected from the offered service, and defines compensatory schemas in case the provider violates these contract pre-defined agreements. Some examples of these agreements could be 99% of availability, resolution of incidents under a specified period of time, data security, etc.

Privacy Policies

[17]Provides an excellent definition of the term policy, which will be very recurrent throughout this document. A policy is a statement of intentions, implemented as protocols or procedures, to guide decisions and achieve a desired goal. When applied under the privacy topic, these policies represent legal documents that explains and discloses the way a party gathers, uses, discloses and manages customer's data (e.g. name, address, medical history, financial records, commercial transactions, etc.). It also informs the client about what information is collected and whether it is kept confidential, shared with partners or sold to other firms of enterprises. These aspects are

usually collected under a written document which outlines rules, provides principles that guide actions, sets roles and responsibilities, reflect values and beliefs and state a protocol of actions.

In this sense, privacy policies forge the individual's physical and moral autonomy, so a set of laws in different countries have been developed over the years to protect and enforce these policies, although there exists differences in the way these privacy laws are implemented and enforced in different countries. While the European Union enforces that data protection laws must be met by any business operating or transferring personal information about any EU citizen or business, in United States privacy laws only apply to the public sector, not the private sector.

Terms and conditions

Also known as Terms of Service (ToS), they are rules which an entity must agree and accept in order to use a service. The terms-of-service agreement is mainly used for legal purposes by websites and Internet service providers that store a user's personal data (e.g. e-commerce sites and social networking services). A legitimate terms-of-service agreement is legally binding, and may be subject to change.

A terms-of-service agreement typically contains sections explaining the user rights and responsibilities of expected usage and potential misuse of the service; accountability for actions, behavior, and conduct; a privacy policy outlining the use of personal data; possible payment details such as membership or subscription fees, etc.; possibly a policy describing procedure for account termination; disclaimer/limitation of Liability clarifying the service's legal liability for damages incurred by users; and user notification when the terms are modified.

2.4. Related Technologies of cloud technology

[18]Cloud computing is often compared to the following technologies, each of which shares certain aspects with cloud computing:

1. Grid Computing

Grid computing is a distributed computing paradigm that coordinates networked resources to achieve a common computational objective. The development of Grid computing was originally driven by scientific applications which are usually computation-intensive.

Cloud computing is similar to Grid computing in that it also employs distributed resources to achieve application-level objectives. However, cloud computing takes one step further by leveraging virtualization technologies at multiple levels (hardware and application platform) to realize resource sharing and dynamic resource provisioning.

2. Utility Computing

Utility computing represents the model of providing resources on-demand and charging customers based on usage rather than a flat rate. Cloud computing can be perceived as a realization of utility computing. It adopts a utility-based pricing scheme entirely for economic reasons. With on-demand resource provisioning and utility based pricing, service providers can truly maximize resource utilization and minimize their operating costs.

3. Virtualization

Virtualization is a technique for hiding the physical characteristics of computational resources. This means that a single physical resource, such as a server device storage or operating system (OS), is seen as multiple logical resources. In essence, this consists of the imitation of behavior of one resource by another. Virtualization can also be defined as a method for sharing multiple computational resources in isolated environments, known as virtual machines, by applying

concepts of partitioning, timesharing, partial or full machine simulation, emulation, and QoS, among others.

Virtualization is recommended to consolidate multiple servers into one host; isolate different user applications in a single host; run/debug software and OS built for one architecture under another, while simplifying the installation of infrastructure software in different areas; and test applications in non-existing hardware.

Theoretically, virtualization types can be grouped as follows:

Full Virtualization: A software layer provides a generic hardware abstraction without the need to modify the OS running on VM. Access control to system physical resources is managed by a hypervisor as soon as the translation of non-privileged instructions to access physical devices such as disk, memory, peripherals, and others is performed by the virtualized OS. Virtual Box is an example.

Para virtualization: The Guest OS is modified to interact directly with hypervisor drivers and gain direct access to hardware routines. Recent advances in CPU hardware have shown para-virtualization to be as efficient as full virtualization. Para-virtualization is an alternative to gain access to resources available in hardware, instead of using generic abstraction instructions provided by full virtualization. In para-virtualized systems, only the OS in domain 0 (zero) has access to devices in a privileged way. VMs in levels above zero field have access to devices through the VM from this domain 0. Xen and UML are examples.

Hardware Virtualization: Virtualization in hardware reduces the need to use a para-virtualized OS to provide direct access to hardware resources. Hardware virtualization is offered with a

virtualized software environment modified to interact directly with the hardware, usually the processor. KVM is an example of extension of the Linux kernel to provide hardware virtualization.

Virtualization in OS level: Implementation of multiple isolated execution environments within a single OS kernel. This approach allows a performance close to native and offers features of dynamic resources management; however, it is not possible to execute different kernels at the same time. LXC, OpenVZ, and Linux VServer are examples.

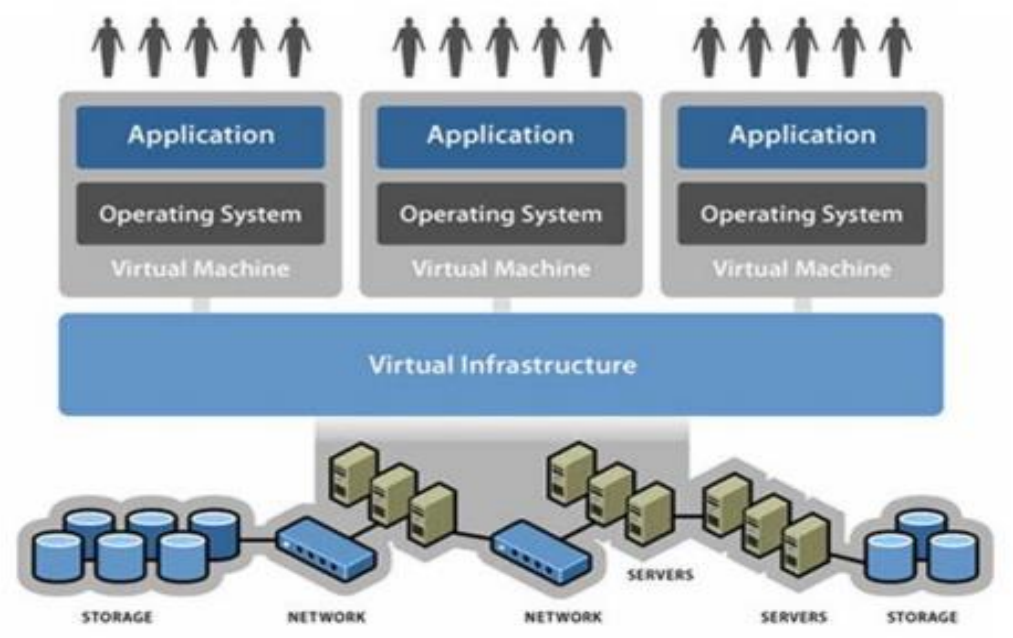

Figure 5: virtualization mechanisms *[19]*

Advantages of virtualization in cloud computing

The most important technology that makes cloud computing possible. Virtualization technologies abstracts physical computing resources (CPU, storage, network, memory, databases...) from applications and end users, providing the capability of creating pooled resources accessible to anything authorized to use them.

This capability provides the multi-tenancy properties of cloud-computing, providing scalable and shared resource platform capabilities for all tenants.

Virtualizations technologies are supported by the hypervisor, a layer of software between an operating system and hardware platform that is used to operate Virtual Machines like launch or terminating, and implements and manages virtual CPU (vCPU), virtual memory (vMemory), event channels and memory shared by the hosted Virtual Machines (VMs), controls data Input/output and memory access to devices. When talking about virtualization applications, common users think on applications to create guest OS Virtual Machines like VMWare or VirtualBox, creating instances of Operating Systems that share resources with the host physical machine and other Virtual Machines. However virtualization does not limit to operating systems, but other components that can be virtualized, like storage (Storage Area Networks), databases, and software (Apache Tomcat, WebSphere, Jboss...). Depending on the delivery model, virtualization appears at various layers. In IaaS commonly storage and OS is virtualized, while on SaaS and PaaS databases and software are the typically virtualized component. [19]

Virtualization technology makes cloud computing environment easily to manage the resources. It abstracts and isolates the underlying hardware, and networking resources in a single hosting environment. It increases the security of cloud computing by protecting both the integrity on guest virtual machine and cloud components virtualized machines can be scaled up or down on demand and can provide reliability. It provides resource sharing, high utilization of pooled resources, rapid provisioning, and workload isolation. The recent trends in virtualization are consolidation of data centers thus reducing the managing cost. Apart of its benefits it has some drawbacks like managing virtual resources is critical and migrating services of these resources are difficult in achieving high availability. If one server fail VM will be restarted on the other virtualized server in resource pool

restoring the required services with minimum service interruption. Virtual resources are critical for managing and data monitoring. Running applications with high utilization and availability is a challenging issue. Hypervisor: A hypervisor is a software, hardware or a firmware that provides virtual partitioning capabilities which runs directly on hardware. It is defined as the virtual machine manager which allows multiple operating systems to run on a system at a time providing resources to each OS without any interaction. Hypervisor controls all the guest systems. As the operating system number increases managing is difficult these leads to security issues. If a hacker gets control over the hypervisor he can control the guest systems by knowing the behavior of the system which causes data processing damage. Advanced protection system is to be developed to monitor the activities of the guest Virtual machine. [20]

2.2. Related works

2.2.1. Cloud based Plagiarism detection

There are many papers which deals with plagiarism detection as a broad concept [21]. In case of plagiarism detection using cloud technology, some of the most related papers are reviewed under these section.

[1]Propose the Architectural model for detecting plagiarism in natural language text and presented the analysis of various detection processes followed for effective plagiarism detection. These method is more detailed and comprehensive and further bifurcated parsing techniques where sentence and word chunking is performed on the basis of the rearrangement of the text or word pattern. This software is generally web based which actually have large repository of the text documents in the form of the journal, Research papers, Books etc. They regularly update their

databases to include more and more latest papers. Their approach to plagiarism detection is through the knowledge of key challenges faced during detection, Present plagiarism detecting tool in the market, Experimentation and verification of the techniques and selecting the best out of the available options. Plagiarism Detection Mechanisms can broadly classified as the Web based and Desktop based. The first category classifies the plagiarism detection through web interface where the fed document is searched on the internet to find the possible clues for the plagiarism. The document searching is done through the Document cycle generator, while later is plagiarism detection mechanism through a standalone application on the computer where two documents are compared with each other to possibly locate the percentage of the plagiarism or simply document similarity. They consider plagiarism detection mechanism as document analysis problem that can be understood through the knowledge of the following aspects as detecting plagiarism in natural language is very difficult to locate. Detecting plagiarism in natural language can be better understood through the knowledge of the language issues related to it such as the writing style of any author is it unique and it depends upon author to author, this can be standardized to some extend but not generalized for every case, other issues such as lack of information, changing the words or paraphrasing and lack of standardization in common definition for plagiarism.

CodeAliker [6] discusses on defining a precise definition of plagiarized computer code, various solutions available for detecting plagiarism and building a cloud platform for plagiarism disclosure. 'CodeAliker', these application thus developed automates the submission of assignments and the review process associated for essay text as well as computer code. The most interesting part of these research is to build a cloud application for the engine. Ruby on Rails, a full stack framework for Ruby is excellent for agile development and sustainable productivity. It boasts a high modular design, excellent package management capabilities, database abstraction

with ORM (Object Relational Mapping) library and ease of deployment. The application is built with the MVC (Model – View – Controller) design pattern inherent on the Rails Framework. For plagiarism detection in a university or an academic institute the needs are critical and point towards the cloud. The computing requirements are thus addressed. Also these system is centralized and easily scaled.

[5]Explains how collaborative efforts in terms of technology and content, can help improve plagiarism detection and prevention. It presents a web service oriented architecture, which utilizes the collective strength of various search engines, context matching algorithms and indexing contributed by users. The proposed framework is an open source tool, yet it is extremely efficient and effective in identifying plagiarism instances. By creatively using distributed processing capabilities of web services, this tool offers a comprehensive mechanism to identify pirated contents. With an aim to extend current plagiarism detection facilities, the proposed framework not only tries to reduce known deficiencies but also aims to provide plagiarism protection mechanism. The distributed indexing approach adapted in the system provides scalability to examine deep web resources. Network nodes with more focused indexing can help build domain specific archives, providing means of context aware search for semantic analysis. These paper concentrates on the service oriented collaborative architecture.

The process of detecting plagiarism includes the following steps:

1. The submitted document is broken down into moderately sized text chunks also called fingerprints. This document source can also be marked for indexing in the local database, accessible via a collaborative search API. System allows the flexibility of configuring automatic fingerprint generation. The text chunks can be based on sentences or sequential word group limited by a character length. The existing prototype uses text segmentation based on character length.

System split text at word boundaries (spaces) at or before specified number of characters limit. In experiments with prototype a fingerprint length of 200 characters is used. This particular size is selected because common web search APIs produce a matching result snippet/description segment of maximum 255 characters including some joining characters.

The fingerprint size can be changed in accordance with search-indexing parameters and document size. Furthermore the text segment can be refined using stop word removal and/or text normalization process.

2. The plagiarism check is initiated by querying the internet using the fingerprint data. The selected APIs search the web. Locally indexed document sources (signature in more abstract form) and that of peer nodes can also be queried if collaborative search service is enabled.

3. Most relevant matches obtained via the search services are passed to similarity analysis service. The existing active service uses word vector based similarity detection as described earlier.

4. Fingerprint similarity scores of a document are calculated to find the plagiarism percentage. The document text linked with the similarity scores, matching contents and source links, is presented to the user within a final report. Collaborative web service oriented architecture substantially extends current plagiarism detection systems. With flexible and extendable services, rich web user interface, standardized XML based inter application communication and collaborative authoring, it brings us a step closer towards Web 2.0 applications.

[22]This paper investigates, utilizing a literature review, how plagiarism detection metrics and a framework for providing effective feedback to students and educators could be implemented to enhance the teaching and learning processes.

The predominant technique used for detecting plagiarism is to evaluate how a piece of source code was constructed over time. By analyzing the students' programming patterns, lectures can

be adapted to address problem areas and react accordingly. The paper also provides an overview of current metrics used for plagiarism detection and suggests ways of improving the process by including enhanced techniques for the gathering of metrics over time as well as suggesting ways to use the metrics to aid learning on all cognitive levels.

[4]Their proposed detection process is based on natural language by comparing documents. A similarity score is determined for each pair of documents which match significantly. They have implemented SCAM (Standard Copy Analysis Mechanism) which is a relative measure to detect overlap by making comparison on a set of words that are common between test document and registered document. Their plagiarism detection system, like many Information Retrieval systems, is evaluated with metrics of precision and recall.

The steps followed to achieve a detection system. There are four main steps:

Indexing the dataset: The dataset is composed of several documents which are preprocessed and indexed. Processing the test document: The test document given in input is processed in order to tokenize it in a list of words; stemming and stop-word removal are applied.

Searching on the index: The index is questioned in order to retrieve a match between the test document and the documents belonging to the dataset.

Evaluating similarity: Using SCAM formula, a similarity measure is calculated between test document and the documents belonging to the dataset.

2.2.2. Types of Plagiarism detection

There are two types of plagiarisms in the broad. These plagiarism types are defined as textual plagiarism and source code plagiarism. Both types of the plagiarism are discussed below.

1. Textual plagiarism

The textual plagiarism is the type of plagiarism in which documents or researches are identical to the original document, scientific paper, journal papers and reports.

2. Source code plagiarism

A source code plagiarism is the other plagiarism type in which students in higher educations copying the whole or the part of someone's original programming source codes.

2.2.3. Document Comparison Plagiarism Detection Methods/techniques

For the time being the researchers' main intention is focus on the textual plagiarism detection aspect, rather the source code plagiarism detection type. So we will focus only on the textual plagiarism detection approach. In textual based plagiarism detection category there are many plagiarism detection methods used. These portion will discuss those textual based plagiarism detection methods. [23]

A. Grammar based method

The grammar based method concentrates on the grammatical structure of documents. Grammar based method uses a string matching approach to detect and to measure similarity between documents. The limitation of these method is unable to detect modified copied text by rewriting some words that has similar meaning. But it is suitable to detect plagiarisms which is copied without any modification. [24]

B. Semantics based method

The semantics based plagiarism detection method focuses on detecting similarities between documents based on the vector space model. It considers the redundancy of the word in the document. In order to find out the similarity they used fingerprint for each document for matching with the finger print of the other document. It is a non partial plagiarism detection to mean it uses

the whole document. The limitation of these plagiarism detection method is it is difficult to fix the part of the plagiarized text in the original document. [24]

C. Grammar semantics hybrid method

These plagiarism detection method is the most important method in detecting plagiarism in natural language. The base of the grammar plagiarism detection technique is natural language processing. It fills the gaps of the above two plagiarism detection methods, grammar based method and semantics based method. It can identify the location of the plagiarized section of the document and can detect the copied texts including modifications with the same meaning.

D. External plagiarism detection method

These plagiarism detection method is similar with textual information retrieval. The information retrieval system returns the ranked set of documents from the database which best matches the query. The external plagiarism detection method uses the inverted index structure. It uses to find the duplications or near duplication documents.

E. Clustering based plagiarism detection

It uses the grammar based plagiarism detection technique in which three steps are followed. The first step is called preselecting, the second step locating and finally the third step is post processing which deals with some merging errors. These plagiarism detection method is used by information retrieval in many purposes. It reduces the searching time of the document. Two traditional clustering algorithms are implemented with document representation based on winnowing fingerprints by the adapting the similarity measures for working with multi-sets and designed a new way of centroid computations.

F. Cross lingual plagiarism detection

These plagiarism detection method is designed to detect suspected documents plagiarized from other language sources. These plagiarism detection approach needs the construction of the cross lingual corpus. [25]

G. Citation based plagiarism detection

These plagiarism detection technique use for categorizing research documents that were read and used without mentioning or referred to those documents. Since it focuses on the detection f semantic content in the citation used in a text research document, it's a part of semantic plagiarism detection technique. It mainly focuses in identifying similar patterns in the citation sequence of academic works for similarity computation.

H. Character based plagiarism detection

These plagiarism detection method has two subtypes. The first one fingerprinting and the second one is string matching. In the finger printing method the preprocessing step involves the creating representative digests of documents by selecting a set of multiple substrings using n-grams from the documents. These representative digests are called fingerprints. The document passages which are suspected as plagiarized are compared to the reference corpus based on their computed fingerprints. Duplicate and near duplicate passages are assumed to have similar fingerprints. [25]
Broadly the above plagiarism detection methods all are categorized under two plagiarism detection tasks. There are two plagiarism detection tasks.

I. Structure based plagiarism detection

The structure based plagiarism detection method focuses on the structure of the text in the document such as headers, sections, paragraphs and references. [25]

J. Syntax similarity based detection

The syntax similarity based method is effective in the research field. Syntactical features are manifested in part of speech of phrases and words in different statements. Basic part of speech tags include verbs, nouns, pronouns, adjectives, adverbs, prepositions, conjunctions and interjections.

2.2.4. Plagiarism detection tasks or approaches

Plagiarism detection tasks are categorized under three main tasks. The following section is discussing about the intrinsic, the extrinsic and the hybrid plagiarism detection tasks.

1. Intrinsic plagiarism detection

The intrinsic plagiarism detection method is the plagiarism detecting approach in which without using any reference. Intrinsic plagiarism detection assesses cases of plagiarism by searching into possible suspicious documents in isolation. This type tries to represent the ability of the human to detect plagiarism by examining differing writing styles. Intrinsic plagiarism aims at identifying potential plagiarism by analyzing a document with respect to undeclared changes in writing style. [26]

Detection methods that are applied to one or more documents belong to the same author, and without external sources, are referred as intrinsic plagiarism detection methods. [27]

Intrinsic plagiarism detection technique does not require a reference collection with original documents. This technique only analyzes the suspicious document trying to find changes in the author's writing style.

2. Extrinsic plagiarism detection

They require a reference collection of potential original documents. Extrinsic plagiarism detection evaluates plagiarism on a reference to one or more source documents in the corpus. Extrinsic

plagiarism detection task tries to utilize the capability of the computer to search for similar documents inside a corpus and retrieve possibly plagiarized documents. [28]

This technique is based in a reference corpus of source documents. The suspicious document is compared with all the source documents to find identical or similar text sections. If the comparison is successful we can confirm a plagiarism in the suspicious document and the source document which has been copied from. In our system only the probably plagiarized sentences identified in the previous phase are compared with the reference corpus. This speeds up the process considerably.

3. Hybrid plagiarism detection

A Hybrid Approach is the combination of intrinsic and external detection features. This is more likely to be applied as an improvement to the filtering stage where external detection is used as a filtering strategy, and then intrinsic detection is applied to identify the location of the plagiarized passage, and vice-versa. For external and hybrid approaches, one can distinguish between an online approach and an offline approach. An online approach performs comparisons not only from a local dataset, but also searches the web for texts that may be the original documents. An offline approach is based on detection algorithms to identify evidence of plagiarism within a local text collection.

Monolingual detection a monolingual detection approach treats the suspicious cases and the source cases in the same language. Suspicious cases are derived from the source cases without any changes to the language. Cross-lingual detection a cross-lingual detection approach is needed when the suspicious cases are derived from source cases of different languages. The derived texts are then translated by manual or automatic means. This approach typically requires language generalization as part of the pre-processing stage. [29]

2.2.5. Comparison of Plagiarism detection tools

In these section different plagiarism detection tools used now a days and the approaches and the algorithms used by these different tools as well as the strength and drawbacks of algorithms and approaches used by these plagiarism detection tools are explained as follows.

1. PlagAware

PlagAware is an online application service that used for textual plagiarism detection, which allows and offers some services for the user for example can search, find, analyze and trace plagiarism in the identified topic similar to the topics, PlagAware is a search engine, which is considered as the main element, which is strong in detecting typical contents of given texts. It uses the classical search engine for detecting and scanning plagiarism, and provide different types of report that help the user or the document owner to decide that is his document has been plagiarized or not. The two primary fields for PlagAware plagiarism search engine is webpage monitoring for theft contents and transmitted text assessment. There are three application fields of PlagAware:

Tracing content theft: Webmaster can use PlagAware for detecting and tracing plagiarisms of websites, in order to find out the plagiarized or the copied contents. PlagAware is considered as strong total solution software systems, which allow the operators of websites to do an automatic observation of their own pages against possible content theft.

PlagAware is used in search for plagiarisms of student's academic documents and analyze them. Also it is used to assess plagiarism, also to follow and prove the origin of the works including all of academic documents. It generates report that helps them to fast detection of plagiarism. Proof of authorship is also provided by PlagAware: it became more important to the authors to ensure that the authorization have been granted to their publication including all types of publication this gives them additional competitive advantage and increase the value of your work. [24]

Features of PlagAware plagiarism detection tools

PlagAware is a search engine that allows the user to submit his document and PlagAware start searching over the internet. So mainly it does not have local database but it offers checking other database that are available over the internet. [30]

PlagAware is an online application and it considered as one of search engine, allows the student or webmaster to upload and check their academic documents, homework, and manuscript and articles to be searched against plagiarism over worldwide web. And also provides a webmaster to have capability to do automatic observation of their own page against possible contents theft.

PlagAware: support mainly used in academic filed so it provides checking of most types of submitted publication like homework, manuscript, documents, including, books, articles, magazines, journals, editorial and PDFs.

Synonym and Sentence Structure Checking: PlagAware does not support synonym and sentence structure checking. Multiple Document Comparison: PlagAware offers comparison of multiple documents. Supported Languages: PlagAware supports German as primary language, English and Japanese as secondary languages.

2. iThenticate

iThenticate one of the application or services designed especially for the researchers, authors' publisher and other. It provided by iParadigms that have introduced Turnitin in 1996 to become the online plagiarism detection. It is designed to be used by institutions rather than personal, but lastly they provided a limit service for single plagiarism detection user like master and doctoral students and this allows them to check a single document of up to 25,000 words. So they can use this service to insure or to check their draft thesis whether containing correct citation and content originality. The main features of iThenticate are: [31]

Database Checking: iThenticate used its own database that contain millions of documents like (books, paper, essays, articles and assignments), with a large number of this documents that have been stored in iThenticate database locally, allowing the users who have account to do either online and offline comparison of submitted documents against it and to identify plagiarized content.

Internet Checking: iThenticate, is considered as the first online plagiarism checker that provides live and cached links to websites and database to have extensive internet checking to all submitted documents. This Provides deep internet checking. One more advantage is that it can still check your documents even if a website is no longer online, this include all contents of website like forums, message boards, bulletin boards, blogs, and PDFs etc., all this check is done automatically and in (almost) real-time.

Publications checking: iThenticate offers an online and offline detailed and depth checking most types of publication like documents, including, books, articles, magazines, journals, newspapers, website and PDFs etc.

Synonym & Sentence Structure Checking: Not supported by iThenticate.

Multiple Document Comparison: iThenticate offers two types of document comparison document to document and multiple documents checking against database and also direct source comparison word to word also.

Supported Languages: iThenticate supports more than 30 languages, it mean that it supports most of languages like English, Arabic, Chinese, Japanese, Thai, Korean, Catalan, Croatian, Czech, Danish, Dutch, Finnish, French, German, Hungarian, Italian, Norwegian, Polish, Portuguese, Romanian, Serbian, Slovak, Slovenian, Spanish, Swedish, Greek, Hebrew, Farsi, Russian, and Turkish. [32]

3. PlagScan

PlagScan is online software used for textual plagiarism checker. PlagScan is often used by school and provides different types of account with different features. PlagScan use complex algorithms for checking and analyzing uploaded document for plagiarism detection, based on up-to-date linguistic research. Unique signature extracted from the document's structure that is then compared with PlagScan database and millions of online documents. [33]

PlagScan is a web-based plagiarism detection service that permits the users to either copy and paste researches or upload them to be scanned for checking plagiarism. It checks the document with papers available on the database across the globe. PlagScan offers a green, yellow, or red Plagiarism Level score to any documents scanned to indicate the plagiarism level with the corresponding documents. And it also provides the link for the original document from the web.

PlagScan is online software used for textual plagiarism checker. PlagScan is often used by school and provides different types of account with different features.

PlagScan use complex algorithms for checking and analyzing uploaded document for plagiarism detection, based on up-to-date linguistic research. Unique signature extracted from the document's structure that is then compared with. [34]

PlagScan database and millions of online documents. So PlagScan is able to detect most of plagiarism types either directs copy and paste or words switching, which provides an accurate measurement of the level of plagiarized content in any given documents. The Main features of PlagScan are:

Database Checking: PlagScan it has own database that include millions documents like (paper, articles and assignments), and articles over World Wide Web. So it offers database checking whether locally or others database over the internet.

Internet checking: PlagScan is an online checker so it provides internet checking to all submitted documents. Whether that the document available on the internet or available in the local database or cached.

Publications Checking: PlagScan: is mainly used in academic filed so it provides checking most types of submitted publication like documents, including, books, articles, magazines, journals, newspapers, PDFs online only.

Synonym and Sentence Structure Checking: PlagScan does not support synonym and sentence structure checking but provides Integration via application programming interface in your existing content management system or learning management system possible.

Multiple Document Comparison: CheckForPlagiarism.net offers comparison of multiple documents in parallel.

Supported Languages: PlagScan supports all the language that use the international UTF-8 encoding and all language with Latin or Arabic characters can be checked for plagiarism. [35]

4. CheckForPlagiarism.net

CheckForPlagiarism.net was developed by a team of professional academic people and became one of the best online plagiarism checkers that used to stop or prevention of online plagiarism and minimizes its effects on academic integrity. In order to maximize the accuracy CheckForPlagiarism.net has used the some methods like document fingerprint and document source analysis to protect document against plagiarism. The fingerprint-based approach used to analyze and summarize collection of document and create a kind of fingerprint for it. Some of numerical attributes can be used by fingerprint that somehow reflects in the structure of the document. So by creating fingerprint for each document with some of numerical attributes for each

document in the collection, we can easily find the matching or the similarity between documents across billions of articles.

Using this feature by CheckForPlagiarism.net increased the efficiency in detecting most types of plagiarisms. The main features of CheckForPlagiarism.net are:

Database Checking: CheckForPlagiarism.ne uses its own database that include millions documents like (paper, articles and assignments), and articles over World Wide Web. So it offers fast and reliable depth database checking, also provides checking through all other databases in different fields like medical database, law- related database and other specialty and generalized databases.

Internet Checking: CheckForPlagiarism.ne: live (online) and cached links to websites used for extensive internet checking to all submitted documents. One more advantage is that it can still check your documents against if a website that is no longer online, this include all contents of website like forums, message boards, bulletin boards, blogs, and PDFs etc., all this check is done automatically and in (almost) real-time.

Publications Checking: CheckForPlagiarism.net offers detailed and deep checking of most types of submitted publication documents, including, books, articles, magazines, journals, newspapers, PDFs etc. this is done whether the publications is available online (active on the internet) or not available on the internet offline (store paper based).

Synonym & Sentence Structure Checking: CheckForPlagiarism.net is said to have a sole advantage, that other soft-wares do not support, which is the fact that it uses a ”patented” plagiarism checking approach. In which the sentence structure of a document is checked to ensure improper paragraphing and thus is susceptible to plagiarism. Also a synonym check is done to words and phrases to identify any attempt of plagiarism.

Multiple Document Comparison: CheckForPlagiarism.net can compare a set of different documents simultaneously with other documents and can diagnose different type of plagiarisms at the sometimes.

Supported Languages: CheckForPlagiarism.net supports English languages, Spanish, German, Portuguese, French, Italian, Arabic, Korean, and Chinese languages. [24]

5. PlagiarismDetection.org

PlagiarismDetection.org: an online service provides high level of accuracy result in plagiarism detection. Mainly designed to help the teachers and student to maintain and to ensure or prevent and detect plagiarism against their academic documents. It provides quickly detect plagiarism with high level of accuracy.

The main features of PlagiarismDetection.org:

Database Checking: PlagiarismDetection.org used its own database that contains millions of documents like (books, paper, essays, articles and assignments).

Internet Checking: PlagiarismDetection.org is an online plagiarism detector, so it is mainly based on the internet checking and is faster in plagiarism detection, it does not support offline detection.

Publications Checking: PlagiarismDetection.org offers the students and teachers to check their publication against the published document and support most types of publication.

Synonym & Sentence Structure Checking: PlagiarismDetection.org not supports Synonym & Sentence Structure Checking.

Multiple Document Comparison: PlagiarismDetection.org does not support multiple document comparison but it takes long time to return the result.

Supported Languages: PlagiarismDetection.org supports English languages and all languages that using Latin characters. [24]

6. Turnitin

Turnitin is the leading originality checking and plagiarism prevention service used by many students and faculties, and thousands of higher educations or institutions worldwide. Turnitin encourages best practices for using and citing other people's written resources or researches. These plagiarism detection tool is a tool which follows external plagiarism detection approach, and the semantics technique [28].

7. PlagiarismChecker

PlagiarismChecker plagiarism detection site offers a simple copy-and-paste function that matches the researches against other sites and articles on the web.

8. Unplag

Unplag which is the plagiarism detection tool is especially suggested to students, educators, writers and editors. With Unplag, the user don’t need to install any extra equipment and programs to use the checker, as it operates online. Unplag allows the user to simultaneously scan as many texts of different formats as they need within four a seconds per text-scan with constant in speed.

In order to identify even the smallest plagiarized passage, it checks text against 16 billion pages and documents on Google and Bing. Unplag provides a plagiarism report after each scan is over. This report includes percentage of similarity and originality of the files scanned as well as all highlighted unoriginal elements with hyperlinks to original sources. Additionally, all the files store in the personal account are kept secure thanks to encryption. Pros: Before subscribing, you can test the checker using 275-word free trial. You can create your personal database and make use of four types of scans, which include checking a file against another one or against all the files you have in your account, Unplag database, or the Internet. By default, you can upload .pdf, .docx, .doc, .txt or .html files to be checked for plagiarism. Cons: To use the checker you need to subscribe for one

month, three months or a year (however, the longer the subscription is, the less you pay per month. [36]

The draw backs of the currently working plagiarism detection tools which is extracted from the above discussed plagiarism detection tools. When we see the above discussed plagiarism detection tools, they lack at least one of the features which are described below.

- They are proprietary means they are not accessed freely.
- They are not supporting multiple languages,
- Multiple document comparison
- The Time it takes to check the plagiarism
- The number of characters or word to be allowed at a time is limited
- Some of them are not support the specific text file format
- The algorithms used by those tools are not compatible to detect the text plagiarism in higher educations through the cloud.

So the intended plagiarism detection system which is proposed in this research is suggesting the right plagiarism detection approach and algorithms which will solve the problems of plagiarism especially in Ethiopian higher educations and institutions

Table 2. Comparison of the Plagiarism detection tools *[24]*

Excellent (*****), Very Good (****), Good (***), Acceptable (**)

Features	**PlagAware**	**PlagScan**	**iThenticate**	**CheckForPlagiarism.net**	**Plagiarismdetect.org**
Database Checking	*****	*****	*****	****	****
Internet Checking	*****	*****	*****	*****	*****
Publication Checking	*****	*****	*****	***	***
Multiple document comparison	*****	*****	*****	****	****
Multiple Languages support	*****	*****	*****	*****	****
Sentence structure and synonym checking	****	**	****	*****	**

The table tries to compare and contrast different types of plagiarism tools based on the given parameters or features.

2.2.6. Comparison of plagiarism detection algorithms

In these section we will discuss some of the algorithms used in plagiarism detection tools discussed above. So there are many different algorithms used by the plagiarism detection algorithms.

A. Karp and Rabin's algorithm

Karp and Rabin's algorithm [26] for fast substring matching is apparently the earliest version of fingerprinting based on k-grams. Their problem, which was motivated by string matching problems in genetics, is to find occurrences of a particular string s of length k within a much longer string. The idea is to compare hashes of all k-grams in the long string with a hash of s. However, hashing strings of length k is expensive for large k, so Karp and Rabin propose a "rolling" hash function that allows the hash for the i + 1^{st} k-gram to be computed quickly from the hash of the ith k-gram. Treat a k-gram c1 . . . ck as a k-digit number in some base b. The hash H(c1 . . . ck) of c1 . . . ck is this number:

$$c1 * bk-1 + c2 * bk-2 * . . . + ck-1 * b + ck$$

To compute the hash of the k-gram c2 . . . ck+1, we need only subtract out the high-order digit, multiply by b, and add in the new low order digit. Thus we have the identity:

$$H(c2 . . . ck+1) = (H(c1 . . . ck) - c1 * bk-1) * b + ck+1$$

Since bk−1 is a constant, this allows each subsequent hash to be computed from the previous one with only two additions and two multiplications. Further, this identity holds when addition and multiplication are modulo some value (e.g., the size of the largest representable integer), so this method works well with standard machine arithmetic.

As an aside, this rolling hash function has a weakness. Because the values of the ci are relatively small integers, doing the addition last means that the last character only affects a few of the low-order bits of the hash. A better hash function would have each character ci potentially affect all of the hash's bits. As noted in, it is easy to fix this by multiplying the entire hash of the first k-gram by an additional b and then switching the order of the multiply and add in the incremental step:

$$H'(c2 \ldots ck+1) = ((H'(c1 \ldots ck) - c1 * bk) + ck+1) * b$$

B. All-to-all matching

The first scheme to apply fingerprinting to collections of documents was developed by Manber, who apparently independently discovered Karp-Rabin string matching and applied it to detecting similar files in file systems [12]. Rather than having a single candidate string to search for, in this problem we wish to compare all pairs of k-grams in the collection of documents. The all-to-all nature of this comparison is a key difficulty in document fingerprinting. To illustrate, consider the problem of all-to all matching on ASCII text. Since there is a k-gram for every byte of an ASCII file, and at least 4-byte hashes are needed for most interesting data sets, a naive scheme that selected all hashed k-grams would create an index much larger than the original documents. This is impractical for large document sets, and the obvious next step is to select some subset of the hashes to represent each document. But which hashes should be selected as fingerprints? A simple but incorrect strategy is to select every ith hash of a document, but this is not robust against reordering, insertions and deletions. In fact, prepending one character to a file shifts the positions of all k-grams by one, which means the modified file shares none of its fingerprints with the original. Thus, any effective algorithm for choosing the fingerprints to represent a document cannot rely on the position of the fingerprints within the document. The scheme Manber chose is

to select all hashes that are 0 mod p. In this way fingerprints are chosen independent of their position, and if two documents share a hash that is 0 mod p it is selected in both documents. Manber found this technique worked well. This paper proposed choosing the n smallest hashes of all k-grams of a document as the fingerprints of that document. By fixing the number of hashes per document, the system would be more scalable as large documents have the same number of fingerprints as small documents. This idea was later used to show that it was possible to cluster documents on the Web by similarity. The price for a fixed-size fingerprint set is that only near-copies of entire documents could be detected. Documents of vastly different size could not be meaningfully compared; for example, the fingerprints of a paragraph would probably contain no fingerprints of the book that the paragraph came from. Choosing hashes 0 mod p, on the other hand, generates variable size sets of fingerprints for documents but guarantees that all representative fingerprints for a paragraph would also be selected for the book. Broder classifies these two different approaches to fingerprinting as being able to detect only resemblance between documents or also being able to detect containment between documents. [24]

C. WINNOWING algorithm

The researchers who propose these algorithm, describes and analyzes the winnowing algorithm for selecting fingerprints from hashes of k-grams. We give an upper bound on the performance of winnowing, expressed as a trade-off between the number of fingerprints that must be selected and the shortest match that we are guaranteed to detect. Given a set of documents, we want the find substring matches between them that satisfy two properties: [26]

1. If there is a substring match at least as long as the guarantee threshold, t, then this match is detected, and

2. We do not detect any matches shorter than the noise threshold, k. The constants t and $k \leq t$ are chosen by the user. They avoid matching strings below the noise threshold by considering only hashes of k-grams. The larger k is, the more confident they can be that matches between documents are not coincidental. On the other hand, larger values of k also limit the sensitivity to reordering of document contents, as they cannot detect the relocation of any substring of length less than k. Thus, it is important to choose k to be the minimum value that eliminates coincidental matches

The most common methods of text plagiarism detection and external plagiarism detection approach are of the above. But as we tried to discuss, they have their own short comes in case of the plagiarism detection system proposed in Ethiopian higher educations and institutions context. So better than the above discussed algorithms the n gram algorithm which will be discussed in chapter four has best feature in higher educations plagiarism detection context. The detail about these plagiarism detection algorithm will be find in chapter four.

Chapter Three

3. Methodology

As I have already identified my subject and my research questions, it is now important to match the design and methods with the problem statement and the research questions, in other words, a research strategy. The research strategy is the scientific method that helps answering the research questions. I will first observe an overall presentation of the methodology, with the research methods, the data collection analysis tools, investigating tools, ethics and so on. As mentioned, I have to identify what kind of research questions I have, to choose the right research strategy.

The first thing I considered was the research question. The research questions could be identified with three purposes as explanatory, descriptive or exploratory, the "what" questions leads to exploratory studies and "how" and "why" questions lead to explanatory studies. As my question was to find the answer of does it has a contribution for the community? What deployment model is preferred to be used to detect plagiarism on the cloud? What are the best futures the proposed architecture will have? And is the proposed architecture secured and manageable? So the study I have pursued is exploratory case study. I was aware of the essential variables of the subject, the phenomena, Cloud Computing, plagiarism detection, three tier software architecture, effect of plagiarism and how they affect the researchers.

The question helped me to understand the phenomena in a particular context, Cloud computing architecture and plagiarism detection. Hence, I also intended to get the answer of how the Cloud Computing works, so this lead to exploratory study. Therefore, my question was both exploratory and this lead to Literature review approach. Along with that, as I had no control over behavioral

events and I wanted to focus on contemporary events thus literature review approach was the best approach for my study and I pursued it.

The Extensive review of acknowledged texts, standards documents, industry periodicals and white papers, analysts' reports and conference journals is done in order to have detail understanding on this research work. By applying cloud technology futures, all higher education sectors will form a community cloud and they all are connected through the cloud using virtualization technology.

The Plagiarism detection method suggested in this paper is by comparing the document against a body of documents. This approach can be further divided into two categories; one that operates locally on the client computer and does analysis on local databases of documents or performs internet searches, the other is server based technology where the user uploads the document and the detection processes take place remotely. The result gives a similarity approximation among documents being checked.

In these paper we discuss the methodology for connecting different higher educational institutes which are located in different geographical location to communicate through the cloud for the purpose of having a central database to store their publications through the local database. So that from the input documents establishing the link between them by generating concept of cloud. From the concept clouds generated for a reference document and the plagiarized document we make comparisons to find out the percentage of idea plagiarism in the document.

After having the physical contact with the workers and presenting our work there, the Ethiopian ministry of education has given us some feedbacks about the plagiarism detection which is proposed to be implemented in Ethiopian higher educations context over the cloud. They normally questioned us the general cloud concepts in terms of the proposed plagiarism detection system, the way how to deploy the system, the security issue of the cloud, the cloud management and the

responsibility issues, as well as the hardware and software compatibility issues were the main issues discussed and we incorporate these issues with the proposed system. By explaining the seriousness of the problem in the country, they appreciate the proposed system alongside the problems it solves if implemented especially for Ethiopian higher educations and institutions now a days.

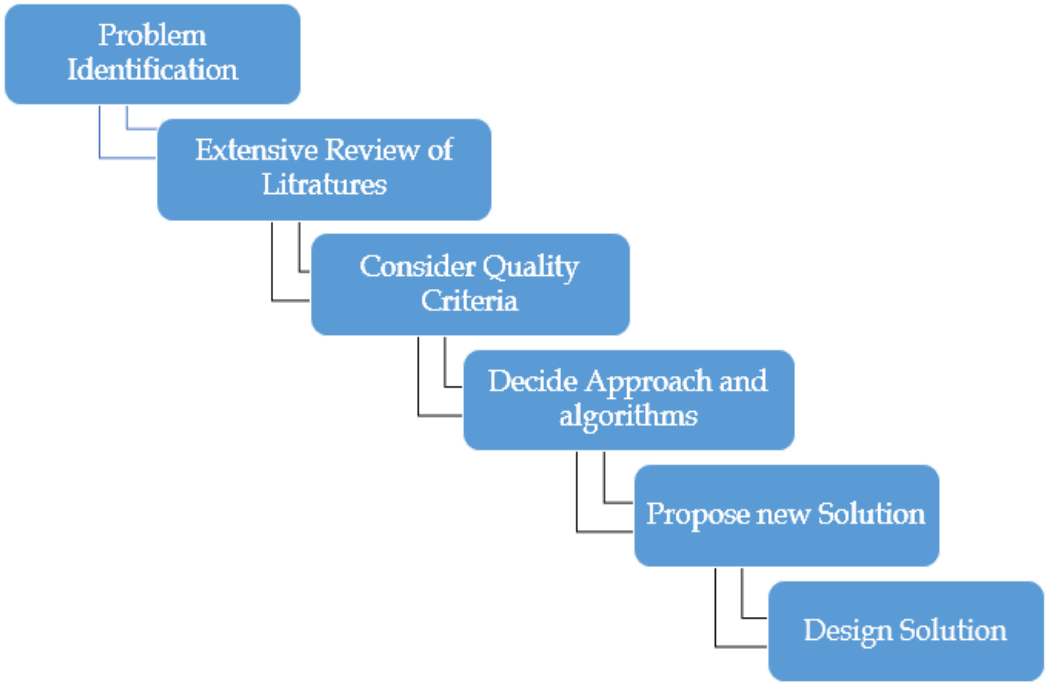

Figure 6. The methods used in the research

Chapter Four

4. Proposed System Design and Analysis

4.1. Introduction

Now a days it is common that Ethiopian higher education students are shared each other the resources of their graduation researches or projects. These is because of the lack of communication or network in between higher educations altogether. The proposed system will solve these communication barrier by developing cloud based plagiarism detection system that will deploy over the cloud so that by using virtualization technology, all higher educations will participate on the system.

The main objective of these study is know that it is to develop a plagiarism detection system that will be deployed over the cloud to fight against plagiarism in higher educations of Ethiopia. And now the detail about the proposed plagiarism detection system will be discussed in these chapter. In these chapter we are going to discuss about the main findings of the study, detail description of the proposed plagiarism detection system, the interaction flow between the user and the system, the model used to detect plagiarism in these system, as well as the user interfaces or prototypes used in these system.

So in order to implement these system, there are many requirements that we have to consider first.

- The effective plagiarism detection methods or approaches that best fits in case of Ethiopian higher educations should be chosen.
- The algorithm used to enhance the similarity detection while checking the similarity ratio should be identified.

- The best implementation tools that has the better performance and that eases implementation of the algorithms used should be selected based on the performance issue.
- The existing higher educations trend to overcome and to check plagiarism suspicious documents.

4.2. Proposed plagiarism detection method description

4.2.1. Document Preprocessing

A document has to pass through several steps before it starts to be compared for plagiarism. Some of these steps are crucial for measuring the overlap between documents. Pre-processing documents is an important stage before measuring their similarities. Main steps involved in document preprocessing are tokenization, stop-word removal, and stemming.

Tokenization: The first step in preprocessing is to parse or clean a document by removing irrelevant information, such as punctuation and numbers, remove capitalization and additional spaces. In general a token is a unit of a document that may be used by a system. For Web documents it is important to remove document markup such as HTML tags, java script functions, etc. before the documents compared.

Stop-word Removal: Stop-words such as “the”, “of” “and”, etc., indicate the structure of a sentence and the relationships between the concepts presented but do not have any meaning on their own and can be safely removed without effecting the accuracy of measuring how similar two documents is.

Stemming: Many words in the English language have multiple variant forms, distinguished by suffix. The suffixes from variant forms can be removed by stemming. Stemming is not essential

step in copy detection but can speed up the process since multiple words are reduced to the same term.

Document Chunking: A procedure of breaking a given document into smaller units (tokens) is called chunking. The chunking procedure is an important issue in any copy detection system since this procedure will influence the accuracy of the system as well as its performance.

4.2.2. Document source comparison method

The document source comparison algorithm is the algorithm chosen to be used in these study. This approach can be further divided into two categories; one that operates locally on the client computer and does analysis on local databases of documents or performs internet searches, the other is server based technology where the user uploads the document and the detection processes take place remotely. The most commonly used techniques in current document source comparison involve word stemming or fingerprinting. In approach moderately sized strings (Fingerprints) from a document are compared for similarities with preprocessed indexes from other documents. The result gives a similarity approximation among documents being checked. The following Figure shows a generic structure of document source comparison based plagiarism detection system.

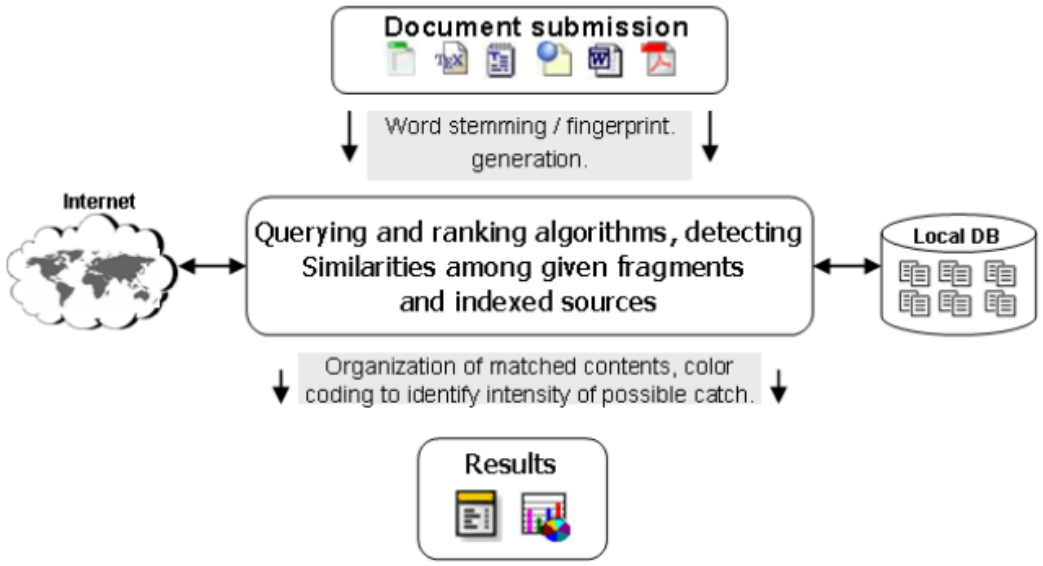

Figure 7: Plagiarism detection with document source comparison *[12]*

In this study the server based technology where the user uploads the document and the detection process takes place on the remote server is the appropriate detection mechanism. So the user from any institution submits the document, that document uploaded to the central database and then through the plagiarism detection SaaS the comparison is made inside the cloud server, and finally the result is displayed for the clients from the higher educations.

These document source plagiarism detection algorithm is deployed on the cloud server and then the SaaS application is accessed by the clients using the User Interface.

4.2.3. Proposed plagiarism detection algorithm

The plagiarism detection method the proposed system will use is based on the automatic plagiarism detection or it is based on computer assisted detection. And the plagiarism detection follows the external plagiarism detection approach. It is to mean that plagiarism is detected by comparing the contents of the submitted research paper with the contents of the papers found in the internet and

publicly available in various databases. It requires a reference corpus to evaluate the request suspected paper.

In external plagiarism detection technique, there are various plagiarism detection methods. But in case of these cloud based plagiarism detection in Ethiopian higher educations, string matching method is employed with n-gram algorithm. Our focus for this research is on a system based on a local database collected from research papers submitted by the local administrator from each higher educations of Ethiopia.

Local Database Based Approaches can be either Structured or Non Structured. The Structured approach creates a graph model of information in the document. This approach is used mostly with source code plagiarism detection. Non-Structured techniques are the most widespread ones and are useful on a wide variety of text plagiarism detection. There are different forms of no structured plagiarism detection techniques. They are classified based on the algorithm they used. String Matching, Document Fingerprinting, and Parameterized Matching are the popular similarity detection algorithms used during plagiarism detection.

Here in our research the plagiarism detection algorithm used is string matching approach engaged with the n gram algorithm.

The scheme of the usual file content comparison system is shown in the following pseudo code:

```
FOR EACH collection file F
FOR EACH collection file G, F ≠ G
Calculate similarity between F and G
```

In case of these plagiarism detection system, we proposed string matching with n grams plagiarism detection algorithm to the Ethiopian higher educations.

In our plagiarism detection system we used Character based plagiarism detection method. The character based plagiarism detection method is the word for word taking of a phrase, sentence, paragraph, or entire source by removing quotation marks and without citing it appropriately. The most prevalent form of plagiarism among higher education students. The Internet has made copy & pasting extremely tempting for students common plagiarism detection techniques rely on character-based methods to compare the suspected document with original document. Identical string can be detected either exactly or partially using character matching approaches.

4.2.3.1. String matching approach with n gram algorithm

The simple algorithm based on string comparisons by removing all comments, Ignore all blanks, and extra lines, except when needed as delimiters, perform a character string comparison between the two files, and maintain a count of percentages of character correlation. This algorithm is run for all possible program pairs. This simple algorithm will detect many cases of plagiarism.

The n gram algorithm is the algorithm used to improve the string matching approach. The calculation of overlapping n-grams, is a common approach to measuring similarity between texts. 3-grams is normally applied to shorter texts (from several paragraphs to a few pages), and 5-grams is usually used in longer texts (more than a few pages). An n-gram represents n number of consecutive words. Similarity scores can be computed by counting the matching n-grams between the suspicious and source documents. Overlapping 5-grams will have five tokens in each n-gram. For example:

Source 5-grams: [when this man returned he] [this man returned he brought] [man returned he brought me] [returned he brought me a letter]

Suspicious 5-grams: [when this man returned he] [this man returned he conveyed] [man returned he conveyed me] [returned he conveyed me a]

The following program simply shows the sample n gram source code that used to generate n gram of the given sentences. This code block generates n-grams at a sentence level. The input consists of N (the size of n-gram), sent the sentence and ngramList a place to store the n-grams generated.

```
private static void generateNgrams(int N, String sent, List ngramList)
{
String[] tokens = sent.split("\\s+"); //split sentence into tokens
//GENERATE THE N-GRAMS
for(int k=0; k<(tokens.length-N+1); k++){
String s="";
int start=k;
int end=k+N;
for(int j=start; j<end; j++){
s=s+""+tokens[j];
}
//Add n-gram to a list
ngramList.add(s);
  }
  }//End of method
```

Sum of absolute differences between n gram vectors of both strings. Example:-

```
> qgrams('abcde','abdcde',q=2)
   ab bc cd de dc bd
V1  1  1  1  1  0  0
V2  1  0  1  1  1  1

> stringdist('abcde', 'abdcde', method='qgram', q=2)
[1] 3
```

Here the proposed n gram algorithm is the 5 grams algorithm, which is the fastest and the accurate n gram algorithm to detect plagiarism of multiple page documents in higher educations plagiarism detection system context.

The reason behind selecting the 5 gram plagiarism detection algorithm is:

It is fast enough, it helps to decrease the time it takes to search or to find the similarity between the given documents. So it is effective since it measures or considers 5 words at a time.

From the other types of the plagiarism detection n grams, 5 gram is the most accurate n gram. The probability of finding the accurate similarity ratio is better when we use 5 gram plagiarism detection n gram algorithm.

The following program is the simple java program that indicates particularly how 5 grams similarity algorithm has compute similarity between sentences.

```
import java.util.*;
public class Test {
    public static List<String> ngrams(int n, String str) {
        List<String> ngrams = new ArrayList<String>();
        String[] words = str.split(" ");
        for (int i = 0; i < words.length - n + 1; i++)
            ngrams.add(concat(words, i, i+n));
        return ngrams;
    }
    public static String concat(String[] words, int start, int
end) {
        StringBuilder sb = new StringBuilder();
```

```
        for (int i = start; i < end; i++)
            sb.append((i > start ? " " : "") + words[i]);
        return sb.toString();
    }
    public static void main(String[] args) {
        for (int n = 1; n <= 5; n++) {
            for (String ngram : ngrams(n, "This is my car."))
                System.out.println(ngram);
            System.out.println();
        }
    }
```

4.2.3.2. Similarity measures of the n gram algorithm

The system designed to detect similarity among text documents calculates content similarity among specified documents, after removal of stop words and stemming the terms. The similarity is estimated between the document samples (assignments) using various measures. For n grams the similarity was computed using three well-known similarity measures, namely Cosine, Jacquard, and Dice coefficient. In order to compute the similarity measure of the sentences using n gram, the statement should pass through the following steps.

Step1: Preprocessing, removing stop-words and stemming the remaining word

Step 2: Constructing the grams vocabulary

Step 3: Generating the binary vectors for query sentences and constructing the inverted index

Step 4: computing the similarity

The Jaccard distance measures dissimilarity between sample sets. It is complementary to the Jaccard coefficient (size of the intersection divided by the size of the union of the sample sets) and is obtained by subtracting the Jaccard coefficient from 1. [37]

$$J_\delta(A,B) = 1 - J(A,B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}$$

Dice's Coefficient Dice coefficient similarity measure is defined as twice the number of terms common to compared entities/strings divided by the total number of terms in both tested strings. [37]

$$s = \frac{2n_t}{n_x + n_y}$$

Cosine similarity Cosine similarity is a popular vector based similarity measure in text mining and information retrieval. In this approach compared strings are transformed into vector space so that the Euclidean cosine rule can be used to calculate similarity. This approach is often paired with other approaches to limit the dimensionality of the vector space. [37]

$$similarity = cos(\theta) = \frac{\vec{A} \cdot \vec{B}}{\|\vec{A}\|\|\vec{B}\|}$$

The above stated similarity measures are symmetry similarity measures. Cosine Similarity is a similarity metric that can be used to measure the similarity of two text documents. Documents can be represented by vectors where each attribute represents the frequency of a word that appears in a document. Even though documents may have many attributes, because of the nature of the human

language the word vectors are actually very sparse. So in order to validate the actual performance of the plagiarism detection system, it is possible to ensure using the similarity measuring metrics discussed in the above.

4.3. Tools used to implement the system

For these method of plagiarism detection Software application design decisions are made by considering the general system architecture. The system is a fully web-based sever side program. The LAMP (Linux Apache MySQL PHP) approach was used for the system. Using LAMP is beneficial as the LAMP components are free and open source. PHP is a server side web scripting language which produces HTML for the client browser. PHP has the benefit of placing the workload on the server and not the client computer, as with other scripting languages such as JavaScript. For this plagiarism detector the ability of PHP to handle regular expressions is also useful.

4.3.1. Advantages of PHP

- PHP is a powerful tool for making dynamic and interactive Web pages.
- PHP is a server-side scripting language.
- PHP supports many database management systems (MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC, etc.).
- PHP is open source software.
- PHP is free to download and use.

4.3.2. Why PHP

- PHP runs on different platforms (Windows, Linux, UNIX, etc.).
- PHP is compatible with almost all web servers used today (Apache, IIS, etc.)
- PHP is easy to learn and runs efficiently on the server side.
- PHP is the language used in Moodle programming.

4.4. Interaction flow between the user and the system

The interaction flow diagram explains that the user in one side interacts with the system in the other side. These interaction begins with the login of the user by entering authentication codes given for every higher educations in Ethiopia. After the authentication is validated from the database, the user is whether allowed to enter to the home page or to login again with correct user name and password. Then the user can upload the papers or documents, can access the papers stored in the database. If the user needs to know the similarity between the paper he/she submit or upload and any other similar document in the database, she/he requests the system for plagiarism checking. And then the system responses the similarity of list of documents with the respected similarity percentage. Similarity has its own level which is expressed interns of percentage. And then according to the plagiarism rule these percentage has its own interpretation on the other hand.

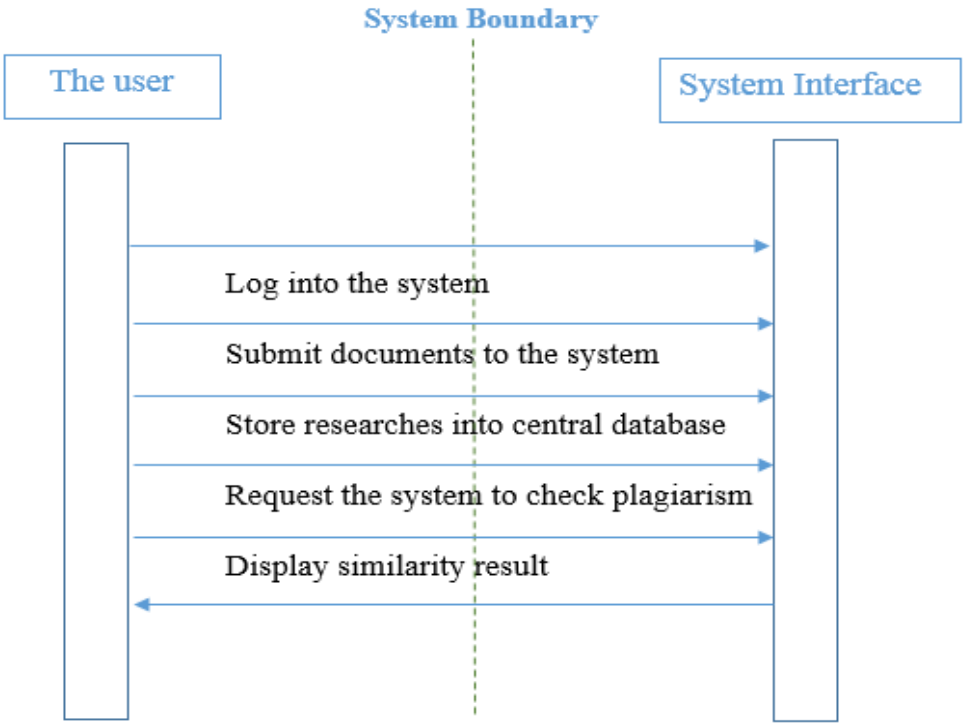

Figure 8: Interaction flow between the user and the system

4.5. Plagiarism detection model

These portion discusses the plagiarism detection cloud model used in these study. And the detail description of the whole cloud based plagiarism detection model stated below the diagram.

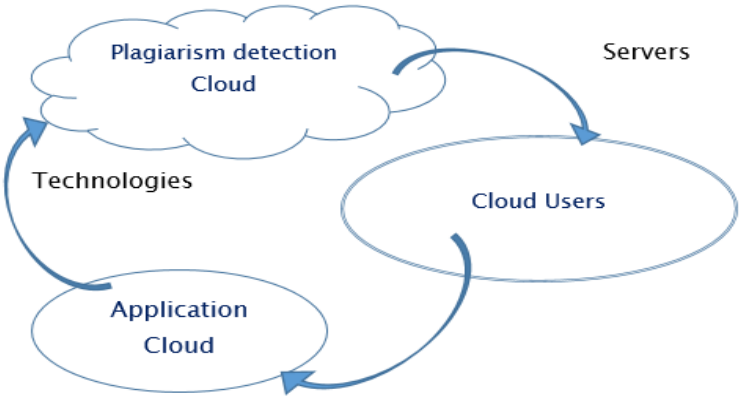

Figure 9: Model of plagiarism detection using cloud

The figure tells us about the model that represents how the system on the cloud can detect plagiarism using the plagiarism detection system which is designed in these thesis.

The cloud users which are from the higher educations will interact using the graphical user interface. They can submit, access and check the documents or resources from the cloud storage which means from the central database.

The application cloud means the application software as a service which is deployed on the cloud performs many different activities. These cloud server tier is responsible to process every queries requested from the user tier.

And finally, the plagiarism detection cloud is the central system that contains the software which processes the document comparison in order to detect the similarity of submitted documents stored in the cloud database.

The overall steps taken by the system is discussed below on the diagram. The documents are uploaded by the user or the administrator of the local server from higher educations to the central database. Through the cloud the uploaded document will be compared with the documents submitted before to the central database. This process takes place at the plagiarism detection cloud server, after execution the result will be responded to the system user via the cloud application interface.

Figure 10: basic steps in the designed plagiarism detection SaaS

The Plagiarism detection SaaS application interface will have the following sample pages.

1. Home page

The admins from the higher educations will login to the home page using there authentication username and password.

2. Upload the papers

The papers will be uploaded to the database and will be stored in, accessed from and checked from the database.

3. Preview paper page

Lastly, the software will display the submitted paper title, author, and ID number, and present two options: go to inbox or submit another paper. Here the user also can search the papers from the database using index or by entering sample paper name at the searching textbox.

4. Successful submission page

All the successfully submitted papers will be listed in the inbox that features the names, titles, similarity index, paper ID numbers, and date of submission.

5. Document viewer with match overview

The Similarity Index rating is important because it shows how much of the paper content matches with the documents in the database. The matched items give the plagiarism checker a clue which part of the paper may have been plagiarized. This feature therefore eliminates the need to execute Internet search (i.e., Google) and manually cross-check passages against generated references. However, high Similarity Index does not automatically translate to a plagiarized paper. There are percepts given after the similarity comparison done between documents.

4.6. Architecture of the proposed plagiarism detection system

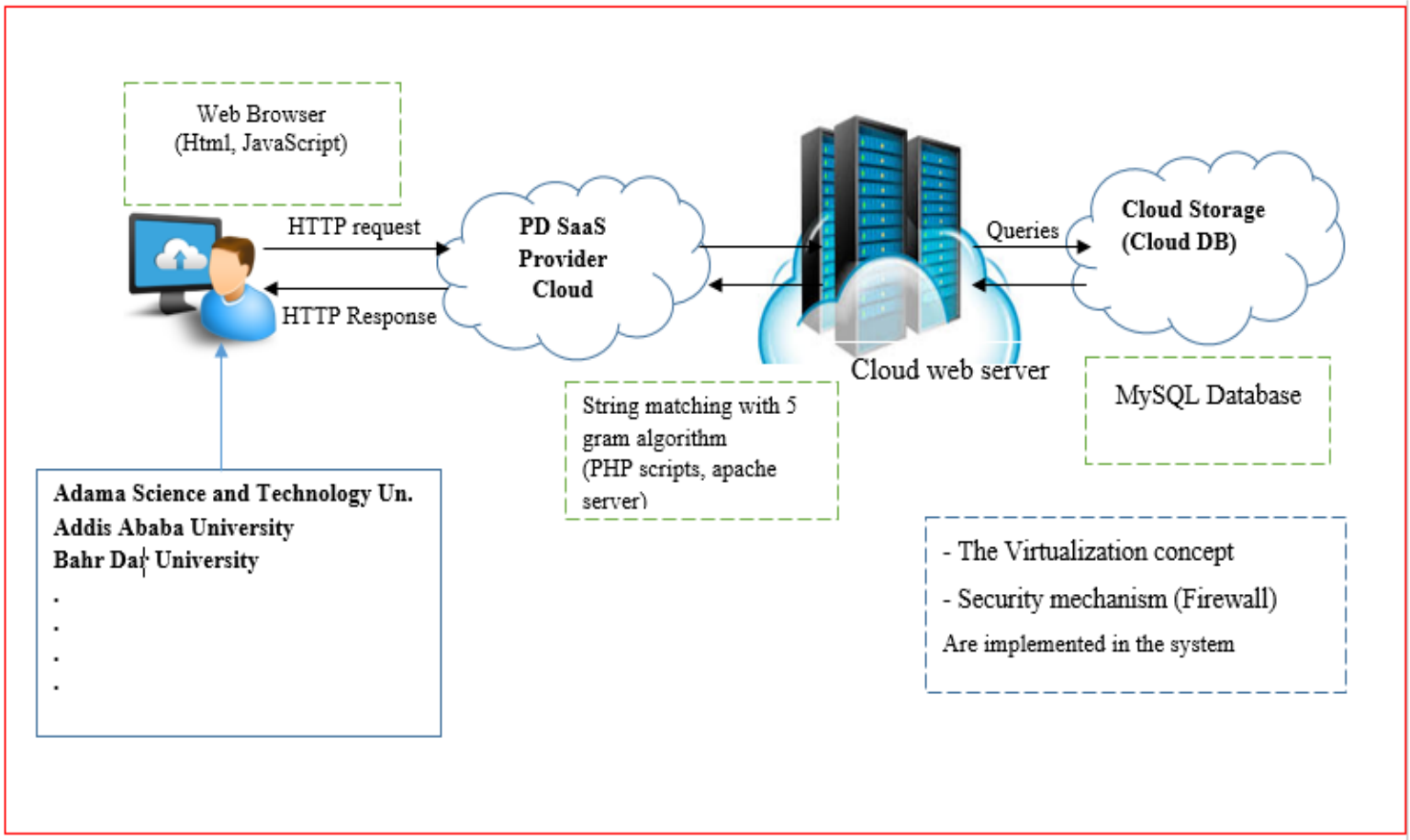

Figure 11. Proposed cloud architecture for plagiarism detection.

Ethiopian Ministry of Education is the cloud service provider in which the plagiarism detection system of Ethiopian higher education service is given to every member higher educations in Ethiopia. So MOE is responsible to control the central cloud server. These is the physical location of the plagiarism detection cloud servers and databases. But by implementing virtualization technology every hardware devices of the plagiarism detection system are accessed and available everywhere at the member higher educations.

The user enters the address in the web browser and in the browser the URL is decoded into protocol/host/file, i.e. host name converted to IP address. Then an issue request is sent to remote server using appropriate protocol (usually HTTP). Also a returned HTML might be accepted.

With the cloud web Server the appropriate action to be taken is identified (insert documents, access documents, comparison), such as fetching a file, or passing request to an interpreter. As such support for thousands for concurrent users, multithreading (allow multiple processor to run concurrently) and caching (holding results in a temporary store to reduce recalculation) is achieved.

On the database server part the interaction with the database is done using standard languages using database specific protocol over TCP/IP. The data structures (for example tables) are defined and modified themselves, that insertion, updating and deleting of data for example. Data maintenance should be maintained with backup and recovered. Access to compilation of queries should be optimized, with indexing or replication of tables.

According to the proposed cloud architecture the cloud database will store the whole documents which are submitted or uploaded from the authorized higher educations which are authenticated as participants on the system. So when the system wants to check similarities according to the documents, it will send the query that will check the similarity between documents.

Redundancy of physical devices is to reduce risks during downtime. Physical devices in the system should be repeated so that when the one fails the other will be functional instead.

Device and network redundancy is like an insurance policy for industrial networks. Acting as a quick-response backup system, the goal of network redundancy is to mitigate the risk of unplanned outages and ensure continuity of operation by instantly responding to and reducing the effects of a point of failure anywhere along the critical data path. When you consider the direct and indirect

costs of unplanned downtime, it becomes clear that making the investment in network and device redundancy is a smart strategy.

4.7. The proposed cloud deployment model

The cloud model used in this plagiarism detection system is Hybrid cloud model. Since higher educations has their own private data's that must be kept secured and since there will be resources which will be accessed with payment, using of the private cloud is recommended. In the other hand resources which will be accessed with free cost and if the resources are not need to be secured they will be deployed on the public cloud. So in order to control these cloud building the hybrid type of cloud deployment model is recommended to incorporate the hybrid features of the two cloud deployment models. Using hybrid cloud deployment model has a great significant for the plagiarism detection system which is proposed to be implemented in Ethiopia higher educations and institutions. It increases the cost effectiveness of the system.

4.8. The proposed cloud service

The cloud service proposed in this plagiarism detection system is Software as a service model, this is because using SaaS has its own best futures in our system. Instead of buying software and then installing it, SaaS users subscribe to the software on a monthly or an annual basis. The relevant applications are used online through files saved in the cloud.

Since the plagiarism detection is already configured, the higher educations has a ready-to-use application. This is not only reduces installation and configuration time, but also cuts down the time wasted on potential problems linked to software deployment. As long as an Internet connection is available, the applications can be accessed from any location.

Since this SaaS plagiarism detection system is cloud-based, it is easily scalable and can be quickly integrated with other similar systems. SaaS enhance the security protection of the plagiarism detection for higher educations of Ethiopia.

Chapter Five

5. Discussion and Result of the study

In these chapter results achieved from the design and analysis chapter, the implementation challenges of the proposed system will be discussed.

The results from the design and analysis chapter shows that using cloud computing technologies is effective method of plagiarism detection. Cloud computing technology so then connects authorized higher educations throughout the country and any plagiarism crime can be empowered and minimized.

The implementation challenges that will face in this system is the possibility of enriching the application to higher educations across the country on the cloud. It needs strong central server management system deployed with the SaaS application and it needs strong Internet connection to access and react with the central server and database. Then the willingness of Higher educations to be a member of these cloud application should be done by the government or in other concerned bodies.

Strong security mechanism also should be maintained because unauthorized users will access the database. Redundancy of servers, network devices should be implemented because uses to avoid downtime problems. Nonfunctional requirement must be considered during deployment.

The proposed plagiarism detection system since uses the Object oriented system approach and the implementation tools are fully open source platforms, it makes the system more effective and manageable.

5.1. Contribution of the system

The proposed plagiarism detection system has its own contribution that involves solutions of the drawbacks discussed in the other plagiarism detection tools. So this research will contribute for higher educations and institutions altogether in Ethiopia. There is no means to control the plagiarism suspicious in Ethiopia now a days. Every individual higher educations may use available plagiarism detection tools online, but it's not common that papers of undergraduate and graduate students of Ethiopian higher educations are deploying the researches online. That is why it makes it simple to use the researches or projects done are available without any modification at another higher education or institution of Ethiopia. Implementing these plagiarism detection system will resolve the problem of cheating ideas or plagiarizing researches done in the other higher educations even in the same academic year. Bring all the researches done in all higher educations and institutions in Ethiopia to the cloud which is deployed and manage by the Ethiopian ministry of education, will make it easy to fight plagiarism. It is possible to access the researches done from the cloud database. The researchers intention is to develop the plagiarism detection system which will support different file formats, that will support to allow unlimited character length, open source plagiarism detection system, that will give fast service and will made a comparison of multiple documents at a time.

Result of the study is based on the conceptual analysis which is the effectiveness of the proposed approaches and algorithms are described on the design and analysis chapter. Since the system is based on the already developed and validated plagiarism detection approaches and algorithms. The main thing here is that just proposing a plagiarism detection system for higher educations and institutions of Ethiopia. And in order to measure the similarity between the documents we already suggest the similarity measuring metrics which are already in practice.

Chapter Six

6. Conclusion and Future work

The primary intention of these cloud based plagiarism detection system is to interconnect higher educations of Ethiopia in order to offer a solution which enables them fighting against plagiarism. In this research work, we fill the gaps of higher educations plagiarism by implementing plagiarism detection system using cloud technology. We have found that plagiarism and cloud Computing are the two very hot topics now days and many researchers are interested on it.

Higher educations since they are the sources of new innovations of researches, researchers finding should be kept patented. So by implementing the cloud technology to deploy the text comparison algorithms of plagiarism detection it is possible to detect the plagiarized resources. Every higher educations should be authenticate through their local cloud server to be involved in the central cloud server so that they can store publications, check plagiarism, as well as use or access those papers stored in the database abroad the country.

So by using plagiarism detection algorithm, implementing cloud software as a service technologies will tackle the problem of plagiarism.

As a future work, by implementing these software as a service cloud based plagiarism detection enabling to detect text, audio and video resource plagiarisms is the proposed system which will be the feature direction for the researchers.

References

[1] G. A. A. A. Amandeep Dhir, "ARCHITECTURAL DESIGNING AND ANALYSIS OF NATURAL LANGUAGE PLAGIARISM DETECTION MECHANISM," *Journal of Theoretical and Applied Information Technology,* pp. 1150-1170, 2005 - 2008.

[2] B. S. A. B.-C. P. R. Martin Potthast, "An Evaluation Framework for Plagiarism Detection," Beijing, Chiina, 2010.

[3] V. G. J. G. Romans Lukashenko, "Computer-Based Plagiarism Detection Methods and Tools: An Overview," 2007.

[4] D. C. F. R. T. D. M. A. H. Daniele Anzelmi, "Plagiarism Detection Based on SCAM Algorithm," Hong Kong, 2011.

[5] B. ZAKA, "Empowering Plagiarism Detection with a Web Services Enabled Collaborative Network," vol. 25, 2009.

[6] R. K. Nitish Upreti, "CodeAliker - Plagiarism Detection on the Cloud," vol. 3, 2012.

[7] B. S. Sven Meyer zu Eissen, "Intrinsic Plagiarism Detection," 2010.

[8] I. R. S. Ivan Balatinac, "Architecting for the Cloud," 2014.

[9] M. K. Asst. Prof. Rajiv Mishra, "Need of Multi-Layer Security in Cloud Computing for on Demand Network Access," vol. 4, no. 6, 2015.

[10] N. J. M. A. B. Amir Mohamed Elamir, "Framework and Architecture for Programming Education Environment as a Cloud Computing Service," Malaysia, 2013.

[11] S. C. Aashita Jain, "E-LEARNING IN THE CLOUD," vol. 2, no. 1, 2013.

[12] G. M. S. A. C. M. Z. H. F. T. J. Shahid Al Noor, "A Proposed Architecture of Cloud Computing for Education System in Bangladesh and the Impact on Current Education system," vol. 10, no. 10, 2010.

[13] M. T. G. K. P. Abhay S. Jadhav, "Survey on Implementation of E-learning System Using Cloud Solutions," vol. 5, no. 2, 2014.

[14] P. D.-D. M. a. T. A. Review, "S.A.Hiremath, M.S.Otari," vol. 1, no. 7, 2014.

[15] P. D. R. K. C. Omprakash Sharma, "Hybrid Cloud Computing with Security Aspect," vol. 4, no. 1, 2015.

[16] Z. Mahmood, Cloud Computing for Enterprise Architectures, New York: Springer, 2011.

[17] "www.baabtra.com," 12 July 2008. [Online]. Available: http//www.baabtra.com.

[18] L. C. R. B. Qi Zhang, "Cloud computing: state-of-the-art and research challenges," 2010.

[19] K. S. R. R. T.Swathi, "VIRTUALIZATION IN CLOUD COMPUTING," vol. 3, no. 5, 2014.

[20] M. S. S. N. K. Dr. Ravish Saggar, "Cloud Computing: Designing Different System Architecture Depending On Real-World Examples," vol. 5, no. 4, 2014.

[21] R. S. B. Prasanth.S, "A Survey on Plagiarism Detection," vol. 86, no. 19, 2014.

[22] A. A. K. B. J. S. v. d. W. Frederik Hattingh, "Presenting an Alternative Source Code Plagiarism Detection Framework for Improving the Teaching and Learning of Programming," vol. 12, 2013.

[23] F. K. B. Z. Hermann Maurer, "Plagiarism - A Survey," *Journal of Universal Computer Science,* vol. 12, 2006.

[24] "Overview and Comparison of Plagiarism Detection Tools," vol. 3, pp. 161-172, 2011.

[25] M. B. L. C. N. M. Ramesh R. Naik, "A Review on Plagiarism Detection Tools," *International Journal of Computer Applications,* vol. 125, p. 0975 – 8887, 2015.

[26] N. S. a. A. A. Ahmed Hamza Osman, "Survey of Text Plagiarism Detection," *Computer Engineering and Applications,* vol. 1, p. 1, June 2012.

[27] A. J. A. MUFTAH, "DOCUMENT PLAGIARISM DETECTION ALGORITHM USING SEMANTIC NETWORKS," pp. 1-164, NOVEMBER 2009.

[28] A. A. F. Wael H. Gomaa, "A Survey of Text Similarity Approaches," *International Journal of Computer Applications* , vol. 68, no. 13, p. 0975 – 8887, 2013.

[29] D. G. Nimisha Singla, "String Matching Algorithms and their Applicability in various Applications," *International Journal of Soft Computing and Engineering (IJSCE),* vol. 1, no. 6, pp. 231-2307, January 29012.

[30] J. O. H. Alan Parker, "Computer Algorithms for Plagiarism Detection," *IEEE TRANSACTIONS ON EDUCATION,* vol. 32, no. 2, 1989.

[31] J. C. Vaughn M. Segers, "An Online System for Plagiarism Detection".

[32] A. K. I. M. O. Yahya A. Abdelrahman, "A SURVEY OF PLAGIARISM DETECTION FOR ARABIC DOCUMENTS," *International Journal of Advanced Computer Technology (IJACT),* vol. 4, no. 6, pp. 2319-7900, 2013.

[33] "A comparison of plagiarism detection tools," no. 0924-3275.

[34] D. G. Nimisha Singla, "String Matching Algorithms and their Applicability in various Applications," *International Journal of Soft Computing and Engineering (IJSCE),* vol. 1, no. 6, January 2012.

[35] A. B. Maria Kashkur, "Research into Plagiarism Cases and Plagiarism Detection Methods," *Scientific Journal of Riga Technical University,* vol. 44, 2010.

[36] M. A. J. R. G. R. a. S. D. MAC Jiffriya, "AntiPlag: Plagiarism Detection on Electronic Submissions of Text Based Assignments," *IEEE 8th International Conference on Industrial and Information Systems,*, pp. 18-20,, august, 2013.

[37] B. Zaka, "heory and Applications of similarity deetction techniques," February, 2009.

[38] J. G. S. W. P. G. Daria Sorokina, "Plagiarism Detection in arXiv," USA, 2010.

[39] C. D. Ranjeet Singh, "Duplicity Detection System for Digital Documents," Vols. Volume-2, no. Issue-5, 2012.

[40] "A Three-Tier System Architecture Design and Development for Hurricane Occurrence Simulation," 2011.

[41] G. M. M. P. Lutz Prechelt, "JPlag: Finding plagiarisms among," March 28, 2000.