ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF INFORMATION STUDIES FOR AFRICA

THE AUTOMATIC EXTRACTION OF BIBLIOGRAPHIC INFORMATION FROM LOCALLY PUBLISHED JOURNALS IN ETHIOPIA: A FEASIBILITY OF OCR

A THESIS SUBMITTED IN PARTIAL FULFILMENT

OF THE REQUIREMENT FOR

THE DEGREE OF MASTER OF SCIENCE IN INFORMATION SCIENCE

BY

ENCHALEW YIFRU

MAY 19, 2000

ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF INFORMATION STUDIES FOR AFRICA

THE AUTOMATIC EXTRACTION OF BIBLIOGRAPHIC INFORMATION FROM LOCALLY PUBLISHED JOURNALS IN ETHIOPIA: A FEASIBILITY OF OCR

BY

ENCHALEW YIFRU

Name and Signature of Members of the Examining Board

Ato Tesfaye Biru, Chairman, Examining Board

Ato Getachew Birru, Advisor

Ato Worku Alemu, Advisor

Dr. John Cowell, External Examiner

THE AUTOMATIC EXTRACTION OF BIBLIOGRAPHIC INFORMATION FROM LOCALLY PUBLISHED JOURNALS IN ETHIOPIA: A FEASIBILITY OF OCR

BY

ENCHALEW YIFRU AYALEW

A thesis submitted to the School of Graduate Studies of Addis Ababa University in partial fulfilment of the requirements for the Degree of Master of Science in Information Science

MAY 19, 2000

ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF INFORMATION STUDIES FOR AFRICA

THE AUTOMATIC EXTRACTION OF BIBLIOGRAPHIC INFORMATION FROM LOCALLY PUBLISHED JOURNALS IN ETHIOPIA: A FEASIBILITY OF OCR

BY

ENCHALEW YIFRU

Name and Signature of Members of the Examining Board

Ato Tesfaye Biru, Chairman, Examining Board

Ato Getachew Birru, Advisor

Ato Worku Alemu, Advisor

Dr. John Cowell, External Examiner

DEDICATION

To Ato Ayalew Filatie and Ato Alemu Gedie's family, particularly my uncle, Ato Sintayehu Ayalew who put a corner stone to resume my discontinued and forgotten! high school education

ACKNOWLEDGMENT

I am deeply indebted to the critical comments made by my advisors Ato Getachew Birru and Ato Worku Alemu, without whose close follow-up my thesis would never have been successful. I appreciate the full co-operation and support made by Ato Dereje Tefferi (SISA staff), particularly in introducing me to the basics of Visual C++. I also acknowledge Ato Workeshet Lamenew (SISA staff) who devoted all his out of office hours to open the SISA computer labs in due course of developing the prototype program.

My special thanks goes to Dr. Taye Tadesse and Dr. Nega Alemayehu who have helped me define the boundary of the research problem at the beginning of my research work.

My deepest appreciation also goes to my best friend Getaw Shumye (Documentation and Information Officer, Agri-Service Ethiopia) for his day to day encouragement and advice and all kinds of sympathy he gave me. I also share my gratitude to my friend Hassen Redwan (Head Librarian at the Derbrezeit Veterinary of Medicine) for his playful moral support.

I have no enough words to express my gratitude to Engida Hailye (Staff at the Department of Business Education) for his endless support from the beginning to the end of my study period. He has been always ready to do everything he can in due course of my stay at SISA.

I peculiarly provide my thanks to Million Meshesha, my classmate, for his tutorial advice and encouragement. He has been always happy to share what he knows.

I am thankful to the full co-operation made by Ato Birru Dori, Head, Information Systems Coordination and Documentation Department, Ethiopian Science and Technology Commission,

to provide me Fangorn (the software used to generate the ISO 2709 format out of the data produced by EARS).

I also acknowledge the contributions of my family in my education and general upbringing.

I want to thank all SISA staff and all my friends who have contributed in one way or another to the completion of my thesis.

I appreciate the supports made by Jimma University (formerly Jimma Institute of Health Sciences) to sponsor me for my study at SISA. I am deeply indebted to Betelihium Setargachew (library staff at Jimma University) who has been my liaison person for all forms of communication to Jimma University.

Last but not least, I extend my gratitude to the full co-operation made by the staff of NALA, particularly the staff of the Legal Deposit and National Bibliography Team and that of the Technique and Computer Services, who provided me all the necessary information required for my thesis.

Enchalew Yifru

to provide me Fangorn (the software used to generate the ISO 2709 format out of the data produced by EARS).

I also acknowledge the contributions of my family in my education and general upbringing.

I want to thank all SISA staff and all my friends who have contributed in one way or another to the completion of my thesis.

I appreciate the supports made by Jimma University (formerly Jimma Institute of Health Sciences) to sponsor me for my study at SISA. I am deeply indebted to Betelihium Setargachew (library staff at Jimma University) who has been my liaison person for all forms of communication to Jimma University.

Last but not least, I extend my gratitude to the full co-operation made by the staff of NALA, particularly the staff of the Legal Deposit and National Bibliography Team and that of the Technique and Computer Services, who provided me all the necessary information required for my thesis.

Enchalew Yifru

TABLE OF CONTENTS PAGE

TABLE OF CONTENTS PAGE

LIST OF FIGURES AND TABLES

TABLES PAGE

FIGURES

ABSTRACT

Research and development communities use journals as mechanisms of communications among themselves. As the size of research output increases from time to time, however, it was impossible to access each and every report that appeared in journals. Therefore, journal articles have to be indexed to facilitate access and control. The activity of indexing has to be systematic, so that research outputs remain accessible to the scientific community. To achieve this lofty goal, indexing has to be made on regional/national basis to serve as part of the universal bibliographic control of journals.

In order to maintain the goal of collecting and indexing publications produced in the country, Ethiopia has established a bibliographic control centre called the Legal Deposit and National Bibliography Team (LDANBT) which is affiliated to the National Archives and Library Agency(NALA). Unfortunately, the LDANBT has produced a journal index (for article level access) neither in printed nor electronic format. In this thesis, therefore, an attempt has been made to develop programme modules that automatically create electronic records out of OCR text obtained from printed journal article title pages. In doing so, the nature of national bibliographic control with respect to journal articles is discussed. As well, techniques of automatically generating bibliographic records from different printed documents is examined. These techniques mainly consist of document analysis and document understanding, which are based on the geometric and non-geometric features of documents.

For document analysis, two levels of segmentation are used. The first level segmentation divides an input text into four zones (first text zone -- consisting of journal title, volume, issue number, year and page range --, article title, author (s) and author abstract) using white line spacing as the end of a text zone. The second level segmentation degenerates the contents of

the first text zone into journal title, volume, and issue number, year and page range. The results of the two level segmentation algorithms are then considered for field classification (document understanding). Classification of fields is made based on geometric and non-geometric features. The geometric feature zone order is used to label article title, author (s) and author abstract. On the other hand the non-geometric features (different punctuation marks consisting of comma, colon, braces, etc.) serves to label the fields in the first text zone as journal title, volume, issue number, year, and page range. The system is 85.57 % successful in correctly segmenting and labelling bibliographic fields. The recognised fields are converted to ISO 2709 format to export into CDS/ISIS for Windows.

CHAPTER ONE: THE PROBLEM AND ITS APPROACH

1.1. General Background

In today's society, most appropriately named as "post-industrial" or "information society", breakthroughs in information technology (IT) are changing the dimension and structure of every sector of human endeavour: business, industry, etc. It is becoming difficult for contemporary organisations to effectively run their day-to-day business without the use of information technology tools. Thus, IT is becoming the backbone of conducting all sorts of organisational processes, and information is the content (substance) that needs to be manipulated and put into use. Needless to say, people are concerned about IT because of its power to simplify the manipulation of raw data, the outcomes of a process, which is valuable information that serves as a vital resource for organisational success. The following statement clearly elaborates the role of information in the contemporary society.

> *In such a society [information society], capital alone doesn't ensure productivity; information is the key economic resource. Industries are thus becoming more brain intensive. Information, the self-regenerative resource (the other resources being the 3M's: money, material and men) is today the fourth dimension of society (Rao, 1990,P.1).*

The unique nature of this resource is due to the fact that, unless it is made accessible to those who use it, it will perish and become out-of-date. Particularly, when the information is contained in journals, its value deteriorates with time. Hence, those who have such information should make sure that it is accessible to and utilised by appropriate users.

Contemporary practice shows that faster and easier access to any kind of information can be possible if data is made available in machine-readable format. The basic assumption towards having digitised records and documents is the fact that such electronic records would be more accessible to a broader category of users, most appropriately through networking

technologies, and there are new efficiencies to be gained in resource-sharing and for storage (Kuny, 1995), setting aside such obvious benefits as speed of retrieval and saving document storage space.

1.2. Optical Character Recognition and Manual Data Entry for Document Digitisation: A Comparative View

Digitisation mostly refers to the process of converting a paper or film-based document into electronic form (Haigh, 1996). For the sake of simplicity and limit its scope, this research project concentrates on capturing data from paper-based documents. Capturing data from paper can be made in one of two ways (Kuny, 1995): manual data entry and scanning and recognising using OCR.

The first approach (manual data entry) enables to manually key-in data into a computer system. Yet, the productivity of manual data entry method is believed to be limited by human skills; and such work becomes ***less attractive with time*** (Gray, 1977). Gray (1977), further, describes the fact that tight schedules, queuing delays, the shortage of computer operators, high personnel turnover rates, inaccurate data transcription, and generally increasing costs of data entry have created unsatisfactory operating conditions for many data processing users. In this method, it has been estimated that 30% or more of systems costs are spent on computer input of information (Gray, 1977). Because of these apparent limitations of manual data entry, it became an agreed concern that the objective of most data entry methods should be to reduce the number of human intervention to a minimum, since input costs are directly related to the level of manual intervention (Oram & Ragozzino, 1977). Generally keyboard-based data entry is time-consuming, labour-intensive, and most importantly very expensive and prone to error (Saffady, 1983).

Views against the keyboard-based data entry process are not only tailor made towards the inefficient (the time, labour, cost problems associated with it) nature of such a method. There are also concerns about the 'in-humanity' of manual data entry. Arguments do exist regarding why people should worry about activities that should not need human labour. The general issue behind this argument is that computerisation should not have to dehumanise people, i.e., activities that can be done by machines should not be done manually. This is well supported by Kilgour (1969, p.30) who said that "computerisation need not to place requirements on people to perform machine-like tasks which machines, not human beings, should be performing best". Therefore, manual data entry is less acceptable by humans and also less efficient option for the digitisation of printed information.

The second approach, scanning and recognising using OCR, uses scanners to lift data from printed pages, followed by an application of OCR software to build a digital record of printed matter that can be edited or indexed. In most applications that are legible for the OCR method, it is claimed that no typing or re-transcription is required with a consequent improvement in the speed and accuracy of data entry (Gray, 1977). OCR is also advantageous in terms of saving data entry cost. While manual data entry demands the cost of hiring, training, attrition, salary increases, over time and other related employee benefits requiring cost, the OCR system may require costs for only one operator, or only a part-time operator (Gray, 1977). Gray (1977) further comments that one major benefit of using high-speed data entry systems such as optical readers is the improvement in the utilisation of installed computer systems; furthermore, improvements in capabilities, reliability, speed, and cost-performance ratio are helping to make OCR a more widely acceptable method of data entry.

1.3. National Archives and Library Agency (NALA) of Ethiopia

1.3.1. Origin and development

Though the history of the National Library of Ethiopia dates back to 1944, Ethiopia didn't have a library named as "National Library of Ethiopia" nor there existed one performing the purposes of such an institution until 1972 (NALE, 1993). In 1975, the library was named as the National Library of Ethiopia, being a department under the Ministry of Culture and Sports Affairs. As part of the restructuring program, the then government forwarded two proclamations, which made major land marks in the history of the Agency.

1. Proclamation no. 127/1977 gave the Ministry of Culture and Sports Affairs the right to establish National Archives.
2. Proclamation no. 50/1975 authorised the National Library to receive three copies of every relevant material published in Ethiopia.

In 1994, the Agency again underwent restructuring and received the name, Ethiopian National Archives and Library Enterprise. As of June 1999, the Enterprise was re-named as National Archives and Library Agency (NALA) through a government proclamation which stated the powers, duties and obligations of the Agency (FDRE, 1999). The NALA is under the Ministry of Information and Culture.

1.3.2. Objectives of NALA

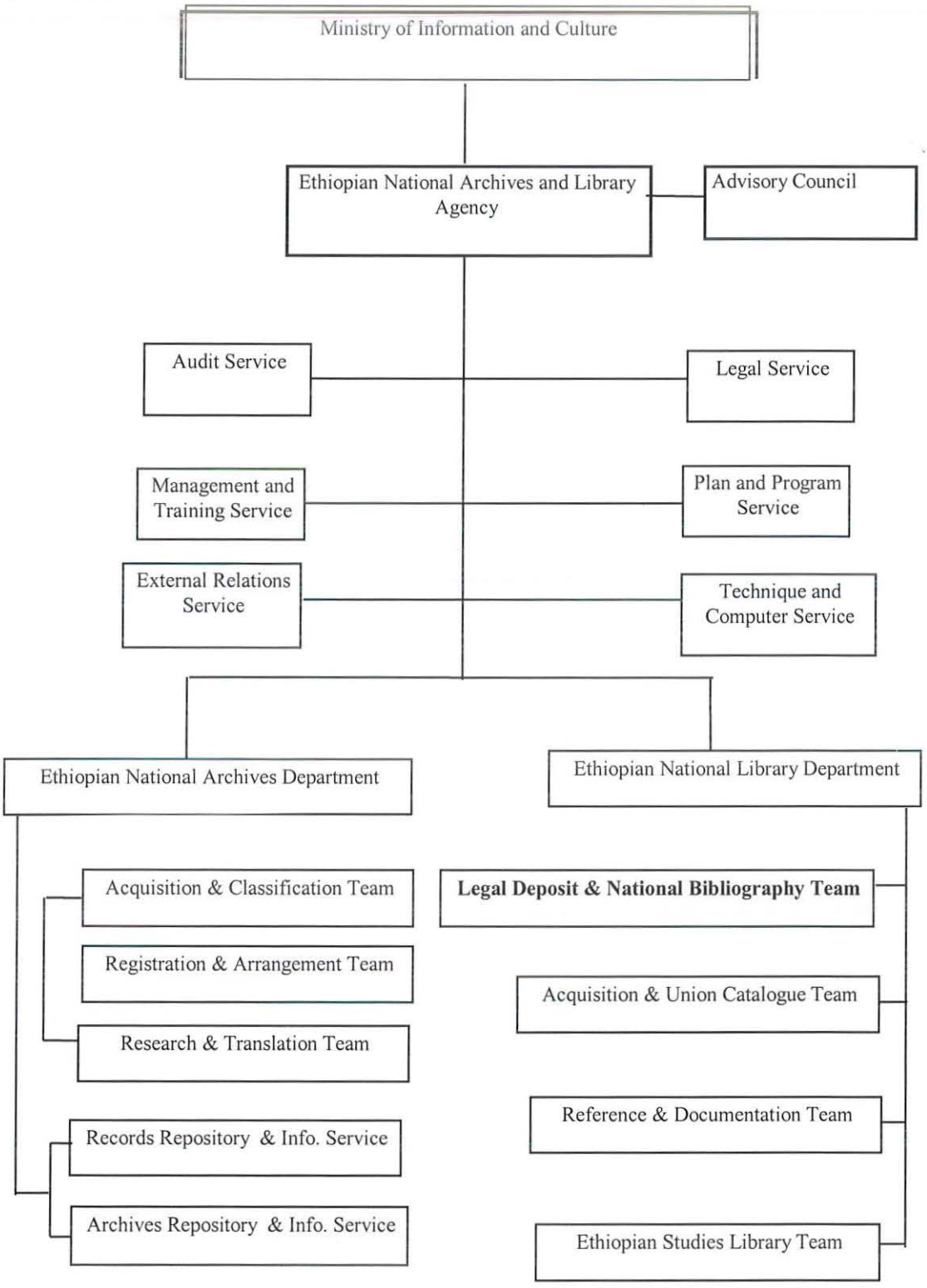
The overall objective of NALA is to collect, systematically organise, preserve, and make the information resources of the country available for study and research purposes. The major functions of NALA are highlighted below.

- Ensure the proper handling and safety of records in the hands of citizens and organisations until their transfer to the agency.

- Serve as a national repository centre for all forms of information sources so that they would be reserved for study and research purpose.
- Establish a record centre, which serves as a temporary storage of records transferred from institutions and citizens, and in which records of significance are appraised to be transferred to the Agency.
- In co-operation with archives, libraries, documentation centres and other information entities, establish, organise and create a database of a national information system which enables an integrated, proper and efficient utilisation of the information resources of the country.
- Nationalise archives which have national importance and which are in possession of individuals, religious institutions, or that are the property of ceased government offices and archives without owner.
- Prepare, publish and distribute the Ethiopian National Bibliography and Periodical Index publications.
- Serve as a national registry centre of ISBN and ISSN of the country.
- Register copy right with regard to works of literature, history and science recorded on different media, which are prepared and printed in the country or printed in the country although prepared abroad.
- Serve as a repository centre for publication of international and national organisations as well as research institutions
- Work in collaboration with state archives and libraries as well as public libraries established or to be established in regional administrations
- Function as a training centre to achieve its objectives
- Work closely with national and international organisations in order to develop and promote professions pertaining to archives and library affairs (FDRE, 1999).

1.3.3. Organisational chart of NALA

In order to meet the above responsibilities, the Agency is organised into two main departments: National Library and National Archives. Both departments are organised into several teams and service units. In addition, there are several administrative and technical units in support of these two main departments and their sub-functional units. The simplified organisational structure of the Agency, showing the horizontal and vertical relationships of the different units is presented in Figure 1.1, next page.



Source: Administration and Finance Service of NALA

Figure 1.1. Organisational Chart of Ethiopian National Archives and Library Agency

1.3.4. Legal Deposit and National Bibliography Team (LDANBT)

1.3.4.1. Objectives

The Legal Deposit and National Bibliography Team is one of the teams under the National Library. The main objectives of the team, as stated in an interview made with the Head of the team (Kebnesh W/Selassie, 9 May 2000), are:

- ensuring the collection and preservation of all forms of publications (printed, audio & video) produced in Ethiopia
- conduct the registration of ISBN and ISSN, though has not yet been materialised
- conducting copyright registration, though has not yet been materialised
- producing national bibliography and periodical index

It was explained that the Team never gives service to users except for materials not available in the Reference and Documentation Team of the National Library. Currently the Team in collaboration with the Technique and Computer Services of the Agency is creating an electronic National Bibliography out of the printed version. In addition, production of visible index for periodicals is in progress since last June 1999. The Head further notes that the Team has the plan of:

- completing the preparation of visible index for periodicals
- microfilming newspapers, due to problem of storage space
- Completing the computerisation of the National Bibliography.
- Designing procedures and functions required for implementing the copyright registration of monographs and serials, and the registration of ISBN and ISSN.

1.3.4.2 Human Resource of LDANBT

Currently, the Team is staffed with 1 MA and 1 Diploma level librarians and 6 support staff (high school grade complete and some years of experience). For an institution dedicated to controlling the nation's publications, such minimal number of staff (both in quality and quantity) is too small to meet the Team's objectives. The absence of a periodical index for journal articles produced in Ethiopia is a direct manifestation of this problem. In such circumstances (setting aside other benefits of automation), the role of automated techniques of producing periodical index for journal articles in Ethiopia would be invaluable.

1.3.4.3. Computing Resource of LDANBT

The Team has one computer (64 MB RAM, 6GB hard disk space, Pentium 2 processor, 350 MHz speed and MS Windows 98 operating system) that is used for the computerisation of the National Bibliography. There are no any other Information Technology resources such as printers, scanners, etc.

1.3.4.4. Number of Locally Published Journals

The "*Ethiopian Periodicals and Non-book Publications*" indices of 1998 and 1999, prepared by the LDANBT, shows that there are about 32 journal titles received by the Team under the Legal Deposit act of 50/1975 (NALE, 1998 & 1999). But, during the physical count of journal issues for five volumes for each journal title, a journal titled "*Ethiopian Trade Journal*" was turned out to be a publication, the contents of which are not journal articles produced based on scientific procedures. Rather, its contents are excerpts obtained from other publications, views of readers, news, etc. Hence, it is excluded from further consideration. Another journal, "*Journal of Dentistry and Oral Surgery*" was not physically available in the

LDANBT's collection. Therefore, there remains to be 30 journal titles produced in Ethiopia and are received by the LDANBT. However, there might be journal titles published in the country but not received by LDANBT. This is "due to the inconsistency of handing over publications to the [Team] ... timely on the part of the printing presses"(NALE, 1998 &1999).

1.4. Technique and Computer Services (TACS)

1.4.1. Introduction

Attempt has been made to identify the level of application of Information Technology (IT) at NALA. In due course, it is realised that major applications are centralised and are under the supervision of Technique and Computer Services (TACS). Therefore, it becomes essential to summarise the objectives, activities, material and manpower resources and future plan of TACS so that it is possible to know the prospect of utilising the computing resources and IT people of TACS by the LDANBT.

1.4.2. Objectives of TACS

TACS was established (having its current name) in 1995. In an interview held with the Head, Dagne Woldie (9 May 2000), the main objective of TACS is to provide such services as binding, microfilm & microfiche production, restoration and conservation of archival materials and computer related applications to the two main departments (National Library and National Archives) of NALA. In order to meet this major objective, TACS is undertaking the following main activities.

- Develop databases using *CDS/ISIS for Windows* for:
    - The National Bibliography

- Accession list of library materials
- Inventory and description of archival materials
- The bibliographic description of Audio-video materials
- Materials obtained from the Internet that are relevant for the National Library and National Archives
- The address list of world wide national libraries, archives and documentation centres.

- Provide major Internet services (e-mail and searching on the WWW) for the community of NALA
- Printing Services for offices with no printers.

Regarding TACS future plans the Head notes that NALA's web-page development and digitising manuscripts are the main activities to be launched soon.

1.4.3. Human Resource of TACS

The following table shows manpower distribution of TACS.

TABLE 1.1. Profile of TACS Personnel

No.	Position	Qualification	Subject Of Study	Number
1	Programmer	B.A.	Library Science	1
2	Data Encoder	Diploma	Accounting & Library Science	2
3.	Computer Operator	High School Complete	Electric And Electronics	1
4.	Microfilm/Microfiche Librarian	B.A.	Library Science	1
5.	Microfilm/Microfiche Producer	High School Complete	Technique School	1
6.	Restorer	Diploma	Art School	1
7.	Binding Staff	High School Complete	Academic School	3
8.	Projector Operator	High School Complete	Technique School	1
9	Secretary	Diploma	Secretarial Science (from Technique School)	1
TOTAL				12

Dagne Woldie (9 May 2000) explained that there are two main IT positions (database administrator and systems analyst) that are supposed to be filled soon.

1.4.4. Computing Resources

NALA has got a total of 11 computers, out of which 6 are distributed over different offices (one of which is the LDANBT). The rest 5 are housed in the TACS. Table 1.2 shows profiles of these computers.

TABLE 1.2. Profile of Computers at NALA

NO.	Quantity	Hard Disk (GB)	RAM (MB)	Processor	Speed (MHz)	Operating system
1.	2	13	64	Pentium 3	400	MSWINDOWS 98
2.	6	6	64	Pentium 2	350	MSWINDOWS 98
3	1	4	32	Pentium	300	MSWINDOWS 95
4	2	1.99	16	486	100	MSWINDOWS 95

There are also 6 printers at NALA. Four of them are HP LaserJet 4000 and the rest two are LQ 2170 Dot Metrix. Two of the Laser printers are in the TACS and the rest two are in the two departments' managers' offices. One of the Dot Metrix printers is in the Library, while the other is in TACS.

In addition, there is one HP ScanJet 6200c scanner. There is also one CD-Writer (though has not yet been used) which will be used to produce back ups for databases.

1.5. Statement of the Problem

Research and development are two-cause and effect phenomenon. The main goal of all kinds of research is to bring about development. However, a one-time research would not practically bring sound development result. Meaningful development would be the outcome of continuous scientific research and investigation. To keep such continuity, any kind of research report should be well documented and preserved for future reference. The conventional strategy of documenting and reporting original research products is to publish them in periodical journals.

The fact that development is the outcome of research motivated the world wide scientific

community to conduct aggressive research and investigation in every sector of human endeavour. This resulted in the production of thousands of journals world-wide. Because journals became too many to keep track of their production, circulation and create access to each article, some kind of solution was necessary. Therefore, journal indexing and abstracting services started to be used as the tools for bibliographic control. These tools provide a response to the question what research result has been reported on a given subject, in a given nation or region, when and who produced it, and in which journal it is reported.

In order to realise this aim, it was necessary to systematise the way the problem is to be solved. Hence, the concept of national bibliographic control that serves as an input to the global attempt for the Universal Bibliographic Control of publications came into being (Line, 1983).

Organisations known as National Bibliographic Service centres were formed as a response to the quest for a body in-charge of National Bibliographic Control. Though there are many activities under taken by bibliographic centres in various parts of the world, the following are identified to be the most common that are handled by the majority (Collinson, 1969, pp.381-382).

1. *Copyright registration of some or all forms of information materials issued with in the territory served*
2. *Formation of union catalogues of such materials*
3. *International exchange and inter-loan of materials, and co-operation*
4. *International co-ordination of bibliographic work and standardisation of documentation.*
5. *Compilation of current and retrospective national bibliographies*
6. *Compilation of current and cumulative indices to the contents of periodicals in a nation*

With the exception of functions 2 and 3, the Legal Deposit and National Bibliography Team of the Ethiopian National Library conducts or plans to conduct all other activities. The

Acquisition and Union Catalogue Teams of the National Library performs functions 2 and 3.

In-terms of their location, it is said that for those which are funded by the government, national bibliographic service centres are usually made part of the national library (Collinson, 1969), since the bibliographic control of a given nation's publications is better realised with the organisation that collects them (Line, 1983).

Traditionally, national bibliographic control is relatively better effective in creating records for books and monographic publications than having the records of articles for published journals in a nation. In other words, "While journal titles may be recorded in the national bibliography, the articles in those journals are not normally indexed" (Line, 1983). The reasons for this are:

> *... many of them are indexed in international systems..., but a great majority of them is not, and the articles in them are totally inaccessible except by browsing through the journals in question. In a country that produces a lot of journals, the magnitude of the task of comprehensive indexing is obviously enormous, but the fact remains that without it national bibliographic coverage is incomplete (Line, 1983, p.231).*

The realities in the national attempt for the bibliographic control of Ethiopian publications reflect this fact. The national bibliographic service centre of Ethiopia, though not named as such, is the Legal Deposit and National Bibliography Team of the National Library. The Head of the Team (Kebnesh W/Selassie, 9 May 2000) showed that since 1979, this Team was producing national bibliographies entitled "*Ethiopian Publications*", for monographs, and "*Ethiopian Periodicals and Non-book Publications*", for periodicals and non-print materials. The extent of detail regarding periodicals is to create bibliographic descriptions (journal title, place and year of publication, volume and issue numbers, if any) for journal titles but not the

articles in them. These bibliographies are prepared for materials collected under the Legal Deposit act of 50/1975, once in two years. However, since last year, according to the Head, the production of these bibliographies is made possible annually.

It is also indicated that though creating article level access to locally published journals is the responsibility of the Team, it was not realised so far for reasons of:

- Budgetary problem
- Limited number of qualified staff

Hence, there exists no way of knowing what research output is produced and published in a certain subject by a given author/organisation in Ethiopia, except trial and error browsing of journals. Practically, there is no any bibliographic tool such as a national journal index for journal articles published in Ethiopia.

As already indicated earlier in this chapter, LDANBT in collaboration with the Technique and Computer Service of NALA have begun automating the National Bibliography. Part of the printed bibliography is digitised and will be completed soon, said both the heads of the Legal Deposit & National Bibliography Team, and that of the Technique and Computer Services.

Therefore, in order to complement this effort there is a need towards digitising the required bibliographic elements for an electronic journal index from printed journals. The difficulty in this regard, however, is how to convert the printed data from journals into electronic format. This, if manually done, involves much manpower cost, will be prone to error, and takes time to convert the bibliographic elements of articles in all journals published in Ethiopia.

The manual data entry involves two stages:

1. A professional should prepare the bibliographic elements of an article (journal title, volume & issue numbers, year of publication, pagination, article title, author(s), and author abstract) in a certain predefined format on paper.
2. Data entry clerk keys in the data prepared in a certain format into a computer.

The process of transcribing the bibliographic elements from the title page of each article in a journal to paper results in a certain amount of error. Keying these transcribed bibliographic records on paper into the computer will have also a significant amount of error. This means that an increase in the frequency of manual intervention in any data processing environment increases error rates and operating costs and reduces efficiency of data processing (Jetform UK Ltd, 1998).

Fortunately enough, however, since the late 1980s, the availability of relatively in-expensive document scanners and computer-based optical character recognition programs have made OCR an attractively priced data-entry methodology (Saffady, 1994). In line with this, a number of studies have been underway in an effort to automatically generate electronic bibliographic records out of printed materials that can serve as an input to a database management or information retrieval systems. Some of these efforts include the development of a complete document understanding system by Tsujimoto and Asada (1992) and Liang et al (1996); electronic monograph catalogue development by Kebede (1997) and Weibel, Oskins and Vizine-Goetz (1989); and the production of electronic journal indices pursued by Harrison, Roos and Thomas (1995) and that of Le et al (1999).

Tsujimoto and Asada (1992) and Liang et al (1996) have tried to develop complete document

understanding systems that classify the different parts of a given document into its component parts such as title, author, headers, footers, subtitles, body, pictures, etc. Kebede (1997) developed a system capable of converting printed catalogues into machine-readable form. Weibel, Oskins and Vizine-Goetz (1989) have made an attempt to capture different bibliographic fields from the title page of books so as to automate the process of monograph cataloguing. Harrison, Roos and Thomas (1995) have conducted a research to develop a system capable of capturing article entries from the table of content of journal issues. Le et al (1999) have designed a system that can handle entries (article title, author (s), affiliation and author abstract) from the title page of journal articles.

The focus of the present research is to extract the building blocks for article entries from the title page of each article for journals published in Ethiopia. Although Le et al (1999) have pursued a research to automatically capture bibliographic fields from the article title page of journals, they have not yet considered such bibliographic fields as journal title, volume, issue number, year of publication and page range. As these fields are the building blocks for a bibliographic citation of an article, Le et al (1999) used manual data entry to complement the automated labelling system they have developed. More over, any of the previously discussed research efforts didn't consider the extraction of bibliographic elements from article title pages of journal issues. Hence, in addition to article title, author (s) and abstract that are considered by Le et al (1999), the present research considers the automatic extraction of journal title, volume, issue number, year of publication and page range (where a given article is written in a journal issue). In the sense, the present research complements the work of Le et al (1999).

1.6. Significance of the Study

The OCR-based data entry coupled with program modules that can automatically label various bibliographic fields would make it possible to develop electronic national journal index for Ethiopia. The electronic journal index would serve for several purposes, the most important being the following:

1. The electronic national index to the periodical articles would be an inventory of published research reports in Ethiopia; it would serve as a window or the sole authoritative source for researchers who need to look at published research in journals in a given field in Ethiopia. Gorman (1983, p.177) shows that "when a scholar wishes to refer to a published Canadian research in his field of interest, he often uses *Canadian: Publication of Canadian Interest Received by the National Library*".

3. It would serve as a guide for resource sharing. This would help the Ethiopian National Library materialise one of its most important functions: the international exchange of materials. The National Library can exchange the electronic indices of national journals (in diskette or CD) so that they pave the way for selecting publications that needs to be obtained either in exchange or inter-loan.

4. Electronic indices for locally published journals in Ethiopia would serve as building blocks for the Universal Bibliographic Control of research published in journals world-wide. It enables the country to easily make available research reports produced locally to the out side world (foreign researchers) in co-ordinated fashion.

The majority of the journals published in Ethiopia are not available in any known journal index database. As it is easier to learn from the preliminary pages of journals, out of thirty journals published in Ethiopia very few are indexed in commercially available databases through individual publisher efforts of these journals. These journals include *Bulletin of the Chemical society of Ethiopia* (indexed *in Chemical Abstracts, Chemistry Citation Index,*

Environmental Abstracts), *Ethiopian Medical Journal* (indexed in *Current Contents* and *Clinical Practice*) *and SINET: Ethiopian Journal of Science* (indexed in *American Mathematical Review*, *BIOSIS* and *Environmental Abstracts*), *Ethiopian Pharmaceutical Journal* (indexed in *Chemical Abstracts*). It is also learned (RPO, AAU, 1994) that the *Journal of Ethiopian Studies* is indexed in *Ulrich's International Periodical Directory*, Directories prepared by UNESCO, OAU and Association of African Universities.

1.7. Objective of the Study

1.7.1. General Objective

The general objective of this research project is to experiment the technical feasibility of automatically extracting the bibliographic elements of journal article contributions in the English language from the title page of each article in printed journals published in Ethiopia using OCR software and post-OCR program modules.

1.7.2. Specific Objectives

The specific objectives of this study are the following.

1. To explore relevant literature in OCR-based automatic journal indexing, particularly descriptive journal indexing
2. To identify journals published in Ethiopia, the contents of, which are articles produced, based on scientific procedures.
3. To identify the layout of each article title page of these journals
4. To categorise the journal titles based on similarities in the layouts of each article title page
5. To write prototype program that could create automatic labels of bibliographic elements (journal title, volume & issue numbers, year of publication, pagination, and article title, author(s) and the author abstract) from OCR'd article title pages of selected journals.

6. To test the newly developed program module, the extent to which it can be able to detect and label the various fields available on the article title page of journals selected for the experiment.
7. To format the output data of the program into the structures of a text retrieval system such as *CDS/ISIS for Windows*.

1.8. Methodology

For the successful completion of this project, a number of data gathering and programming tools and techniques are put into use.

1.8.1. Data Sources

Both primary and secondary sources of data are used to gather information.

1.8.1.1. Primary Data Source (Interview)

Responsible personnel of the Legal Deposit and National Bibliography Team and that of Technique and Computer Services of NALA are interviewed about their objectives, current activities as well as their short-range and long-term plans.

1.8.1.2. Secondary Data Source (Literature Review)

Relevant materials on OCR-based data entry for the production of bibliographic records are reviewed to learn from their successes and failures. Materials regarding the nature of bibliographic control in general and publications about NALA are also examined.

1.8.2. Sampling

The population of this study is journals published in Ethiopia, the contents of which are articles produced based on scientific procedures. All such journals that are received by the LDANBT of NALA are considered as the subjects of this study. However, not all the volumes of all the journals are examined. Rather, the last five volumes of each journal that are received by the LDANBT are physically examined. This limit is due to time and cost problems that forbid the researcher to physically count all volumes for all journals received by LDANBT.

1.8.3. Program Development Tools and Resources

The following software are used in due course of developing the prototype program.

1.8.3.1 Scanning and OCR Software

WordScan 3.0 is used to acquire the image and recognise the text.

1.8.3. 2. Programming Software

The state-of-the-art approach in software/program development is to use programming languages that have graphics capabilities, since they enable to produce easily understandable software by users. In addition, programming languages that have object-oriented capabilities are chosen over other structured /procedural/ languages, since they enable to produce maintainable software. The programming language, which is claimed to fulfil both, object-oriented and graphics facility is Visual C++. Therefore, *Microsoft Visual C++ 6.0* that is available at the SISA lab and which the researcher is familiar with is used to develop the source code for this research project.

1.9. Scope of the Study

The scope of this study is limited to the journals published in Ethiopia and are available at the LDANBT of the national library, the content of which are research out put made based on scientific procedures. Furthermore, only the descriptive part of the automatic indexing process is considered. Thus, serials indexing can be viewed from two sides.

1) **Subject indexing** (keyword determination) that demands analysing the contents of whole article and / or abstract.

2) **Descriptive indexing** *(*simple identification of bibliographic elements for article contribution in a journal issue*)*. Descriptive indexing demands little or no expertise of indexing; and almost all the descriptive building blocks of an article's citation can be obtained from the title page of each article for most journals.

Needless to say, useful indexing system should include both of these functions. To limit the scope of the problem, however, this research project concentrates on the second, i.e., the descriptive indexing component of the serials indexing problem. These descriptive parts may be captured automatically by the OCR-based method as discussed so far. The resulting output from the post-OCR process is formatted to fit to the structure of the *CDS/ISIS for Windows* database.

1.10. Limitations of the Study

The following is the major limitation for the research project.

Within the available time frame for the research project, it is not possible to develop a program that can handle the different journals that are grouped based on layout variations. Groups of journals one or more of journal title, volume, and issue number, year and page range is located at the bottom of article title page are not considered. At the same time

journals with articles that have no abstract are not considered.

1.11. Organization of the Study

The thesis is organised into five chapters. The first chapter deals with the problem and its approach including the pros and cons of manual and OCR-based data entry methods, information regarding the Legal Deposit and National Bibliography Team and the Technique and Computer Services of NALA, the statement of the problem, significance of the study, the methodology used, the limitation and organisation of the study. In the second chapter attempt is made to discuss the various studies conducted in the area of automated bibliographic record production using OCR. The concern of chapter three is to study the nature of journals published in Ethiopia and categorise them based on features such as order and number of bibliographic fields in an article title page, the location of certain fields (journal title, volume, issue number, year and page range),the presence/absence of author abstract and one or more of journal title, volume, issue number, year and page range. This serves as the basis for designing a prototype program in chapter four. Chapter four deals with the algorithms used to develop the program and the test results obtained based on the articles selected for the experiment, and the formatting procedures of the program output into the ISO 2709 format. Finally, chapter five covers the conclusions made based on the findings and recommendations for future research in the journals published in Ethiopia and also actions expected from the LDANBT and TACS in order to materialise the output of the present research.

2.1. Introduction

Taking into account several benefits of OCR, many OCR-based document conversion systems are being developed for a variety of document related applications. One of these applications is the generation of bibliographic records from printed publications. Predominantly, these applications use a machine-readable text resulting from an OCR-system as input for the production of bibliographic records. When an OCR-system is complemented with program modules that can detect various fields from the freely available text, and automatically label them with field names, a bibliographic record is created that can serve as an input to various Data Base Management or Information Retrieval systems.

Below are a description of some of the efforts made towards the development of OCR-based systems for document conversion and bibliographic record generation.

2.2. The Automatic Extraction of General Document Components

Tsujimoto and Asada (1992) built a complete text reading system capable of detecting the different parts of a given document such as title, abstract, subtitles, paragraphs, etc. The experiment was made on documents taken from magazines, journals, newspapers, books, manuals, letters, and scientific papers.

The system developed consisted of three components: document analysis, document understanding and character segmentation/recognition.

Document Analysis

It is a component that decomposes a document image into several consistent items that represent coherent components of the document, such as text-lines, photographs and other graphics.

During document analysis, a given document image is broken down into component parts to form a geometric structure tree, the nodes of which represent a set of blocks. A geometric structure is a hierarchy of items on a page for modelling the relationships between characters, lines, columns, and the page; it is the geometric relationship between blocks.

The approach is, first to extract words from a document image, which are then merged into text lines. Text lines are then combined into blocks that correspond to various block categories that represent titles, abstract, subtitles, paragraphs, etc. In fact, the word extraction procedure is made the subset of the entire text line extraction process.

There are few steps in the text line and block extraction process.

Segment Extraction

In this step, adjacent connected components (from a document image) are extracted as a segment by connecting two black runs whose spacing (white runs) is shorter than a certain threshold. The threshold is made very small (1mm) so that words located in different columns are not merged with each other.

Classification of Segments

*S*egments are classified into text lines, figures or photographs, tables, frames, etc. according to the physical properties of the segments. These physical properties of a segment are the size (width and height) and aspect ratio (width/height).

Frames are classified further into text and figure frames by examining their contents. If there are lots of text lines inside a frame, this frame is defined as a text frame representing a box. When figures are found inside the frame, it is a figure frame.

Merging Text Segments

The blank length between words is, in most cases, known to be in proportion to the height of the words on a document. Tsujimoto and Asada (1992) applied this concept when merging neighbouring segments which are defined as text lines if the blank space between them is smaller than a threshold which is made proportional to the word's height.

Block Extraction

Adjacent text lines are combined into a block; a threshold defined by the line interval (between text lines) determines adjacency. The concept block is defined the following way. In a document image consisting of several blocks, each block represents a coherent component of the document. And one coherent component corresponds to a set of text lines with the same typeface and consistent line spacing.

As a further classification of blocks, Tsujimoto and Asada (1992) have grouped each coherent document component (i.e., block) as one of *head* or *body* in order to distinguish titles from texts. Head is the name for blocks in which there are only a few text lines; in addition, text is biased to the left or is centred. Larger type fonts are used in head blocks in

many cases. This kind of block corresponds to titles or subtitles when a geometric structure is transformed into a logical structure (a logical structure is the natural relationship between document components). For example, assuming an article title page of a journal document, the article titles precedes the author, abstract, sections and their sub title). Headers, footers, page numbers, and captions also belong to this category. Body corresponds to blocks consisting of text lines only. It normally has many text lines and smaller type fonts. An indentation is often found in the first text line of a body block. Abstracts and paragraphs belong to this category. The result of the procedures described earlier is then represented in a tree structure, where each node of the tree representing a block.

Document Understanding

Document understanding is a component that extracts the logical relationships between blocks. It is a transformation from a geometric structure to a logical structure. A small number of rules for this transformation are in order, based on the general assumption that a document layout is designed according to the human reading manner. The human reading order, for example, assumes that the title precedes the abstract, chapters and sections, while subtitles precede paragraphs.

In fact, the process of document understanding involves attaching a label to each block. The labels include title, abstract, subtitles, paragraphs, header, footer, page number and caption. This is achieved as follows.

In a tree representation of nodes /blocks/ for the geometric structure, if a daughter of the root node has children and it is a head sequence, then that daughter represents a title. If it has no children and it is a head sequence, one of the labels head, footer, page number or caption is

attached to it according to its location on the page. As an example, a block that is centred and located at the bottom of a page is a page number. Any head blocks other than daughters of the root node are subtitles. Body blocks in terminal nodes are normally paragraphs. Body block that is the eldest and whose next sister is a subtitle represents an abstract. A body block with daughters also represents an abstract.

Character Segmentation/Recognition

Since the aim was to develop a complete text understanding system, Tsujmoto and Asada (1992) tried to develop modules that could enable to segment and recognise characters from text lines that are the outputs of document analysis and understanding phases.

Similarly, Liang et al (1996) at the Intelligent Systems Laboratory, University of Washington, have designed a complete document image understanding system that consists of and integrates various components. To design a complete document image understanding system, Liang et al (1996) developed the following main modules: physical layout analysis, text recognition and logical structure analysis modules. In addition, their system has a control unit that is capable of selecting the appropriate modules for the given data at each processing stage.

Regarding the interaction of these modules, given a document image, the page layout analysis module processes the given image and produces segmented homogeneous regions or blocks (such as text, math, figures, etc.). These segmented blocks are then represented by a tree data structure called the 'physical' structure. Each of these blocks is then classified as text or non-text. Those non-text blocks are further classified into various graphical components. The text blocks are passed on to the OCR software to produce ASCII text. The resulting

physical structure and the ASCII text are passed onto a logical structure analysis module. The tasks for the logical structure analysis module include assigning logical labels to text blocks (i.e., text body, section heading, etc.), and non-text blocks (photo, figures, etc., i.e., grouping those non-text entities with their corresponding captions) and so on.

Below is a consolidated version of the physical and logical structure analysis modules developed by Liang et al (1996).

Physical Analysis Module

The logical structure analysis module depends on blocks of text obtained during the physical analysis phase.

The beginning of a text block, such as a paragraph, math zone, section heading, etc. are marked either by changing the justification of the current text line or by putting extra-space between two text lines or by changing the text height. Therefore, block segmentation of text lines is maintained the following way. When a significant change in text line heights, inter-text line spacing, or justification occurs on the input image, it is realised that a new text block begins. Finally, segmented blocks of text are sent to an OCR system for recognition.

Logical Structure Analysis Module

The logical structure extraction procedure refers to the activity of transferring the physical layout structure, described so far, into a specific logical structure consisting of meaningful objects (i.e., paragraph, title, section heading, caption, etc.). The knowledge required for logical labelling is block style knowledge. These include global knowledge about the logical entity location, the local knowledge about the formatting attributes of the logical entity (font size, style), and the spatial relations between logical entities. Consequently, the logical

structure of document image is analysed by labelling entities by corresponding functional labels after hypothesising and testing layout properties of the captured layout structure.

2.3. The Automatic Extraction of Monograph Catalogue Elements

Kebede (1997) produced a prototype system that creates a machine-readable version for the building blocks of a printed card catalogue for the Addis Ababa University Libraries. The system uses the results of scanned and OCR'd card catalogue ASCII texts as input. It predominantly incorporated Anglo-American Cataloguing Rules (AACR2) for the segmentation and classification of various components of a catalogue entry record (call number, author, title, place, publisher, year, pagination, etc.).

Segmentation

It is reported that card images obtained from HP DeskScan scanning software were manually manipulated to create two text zones: call number zone and body zone. Yet, Kebede (1997) admits that this should have been done automatically by the OCR software. Nonetheless, it is reported that the "AUTO ZONE" option of the OCR software created a number of inconsistent zones for varying positions of the call number and body of the card catalogue. Hence, after two zones (call number and body) have been manually created, and unnecessary data like cataloguer code and date of cataloguing have been removed, the card image was sent to the OCR software. In fact, the text conversion was conducted for each zone per card, according to the following order. The call number is always considered as the first zone, then the body as the second zone.

Labelling

To assign fields with their logical labels, card catalogues are first classified into three groups,

based on whether the personal author, corporate author or the title is used as the main entry. After this, ASCII text resulting from OCR software was parsed as follows.

In order to identify and label the call number, input data from the ASCII file is read until an empty line is encountered or five text lines are read. Five text lines are used as a threshold because the maximum number of lines that a manually zoned call number text could take is five lines.

Example:-

RB
37
.C54
1984
Ref.

The rest of the fields are extracted by using:

- AACR2 punctuation marks covering comma, colon, full stop-space-dash, etc. and certain key terms like 'cm.', 'ISBN', '1.', '2.', etc. or
- End-of-file flag

As an example, if a given OCR text record is Personal Author main entry, single character is read at a time from the input file and stored as a character array to a variable 'Surname' (for the author field) until comma (,) is encountered. Then, the content of 'Surname' is written to an output file. As a continuation to this step, character-by-character reading from the input file continue until comma (,) or newline is encountered in order to get the 'forename' of author(s).

The system was 98.33% successful in segmenting and labelling the various components of a catalogue entry.

In another development, Weibel, Oskins and Vizine-Goetz (1989) built a system in order to automate the production of machine-readable catalogue records from the title page of printed books. The system utilises a set of rules (developed based on the analysis of book title page layout) in order to classify the various components of a book's title page (title, statement of responsibility, publisher, place and date of publication and edition statement) resulting from an OCR system. In fact, Weibel, Oskins and Vizine-Goetz (1989) used machine-readable surrogates of book title page images, instead of scanning and applying OCR software.

Regarding the technical implementation of the system, a program called DVI (Device Independent File) parser was used to extract space delimited character strings. This parser assumes that a word (space-delimited string) is formed whenever a wider space between characters is encountered. On top of word (also called tokens) formation, the overall characteristics of the book's title page including left-most and right-most margins, top baseline, bottom baseline, largest font on the page and median vertical gap between printed elements on the page is created.

In order to create compound tokens (functional meaningful units that can be evaluated for their relevance to bibliographic fields) out of space delimited tokens, the following techniques were used.

The space-delimited words and overall page statistics information are represented in Prologue's (Knowledge-based system) internal database for subsequent processing.
Compound tokens (unidentified fields) are constructed by adding tokens to a compound structure until any one of the following three conditions are met:

- Font style changes

- Font size changes or
- Vertical white space separating successive compounds exceeds a specified threshold.

It was realised that these rules result in the grouping of tokens into compounds that have proven useful in delimiting relevant bibliographic fields.

To label these compound tokens as specific fields, some of the rules put into use include font style and size, case class (upper or lower case compounds), page location, keywords, proper name look up, etc. For instance, it was reported that a compound having the largest type face and appearing first on the page is likely to be the book's title; a compound containing the phrase 'edited by' is considered as part of the statement of responsibility; to identify a valid publication date, a search was made for the presence of a four character numeric token. Looking at an external file containing publisher and place names identifies publisher and place names.

Finally, the authors showed that the system was 89% successful in correctly interpreting the required fields.

2.4. The Automatic Extraction of Bibliographic Elements for Journal Article Contributions

A study has been conducted in order to capture bibliographic information for journal article contributions from the content page of journal issues (Harrison, Roos and Thomas, 1995). The report generated by Harrison, Roos and Thomas (1995) described the aims and background of a project, known as the RIDDLE (Rapid Information Display and Dissemination in a Library Environment) project. Accordingly, RIDDLE was meant to experiment the possibility of including catalogue information for journal articles of

individual libraries in Europe to their monograph online catalogue. The need for such an effort was felt because "when scientists wish to perform a literature search in a library catalogue, they will consider the information contained with in a journal article at least as important as a book. In some cases, the article is the more important reference, since it tends to be more up-to-date" (Harrison, Roos and Thomas, 1995, p.15). Moreover, libraries provide content page (current awareness) service that involves sending photocopies of content pages around their users. Hence, it was considered significant to study how such a service could be converted to an electronic procedure.

The entire process of the RIDDLE project consists of procedures like scanning journal issue contents page, capture the text (using OCR software), identify relevant parts of the text (labelling) and load the labelled text to an online library catalogue (OLC).

Conversion of Text Image

OCR technology was used to extract the textual information from scanned images. An important part of the work was the automatic identification of the journal being processed. This knowledge was used to assist the work performed during the subsequent stages. For instance, knowledge about a particular journal was needed to enable the system to automatically classify the different components available in a journal issue title page.

In order to maintain the aim of full automation, ways were investigated and developed for identifying automatically which journal is being considered. The system was tested to capture the following distinguishing features that are proved useful, appearing either on the title page or the first contents page: journal title, ISSN number, bar-code, logo and layout.

The approaches to journal identification that are believed useful are the following.

- Searching for text extraction by OCR software. The text can be a part of the title string or the ISSN number.
- Searching the extracted text for the distinctive pattern of the ISSN number (which is often printed below a bar code as well as appearing on the body of the text).
- Using image recognition to search for an image fragment (e.g., logo).

Tagging the OCR Output

Publishers use different formats to present contents page so that it is possible to clearly identify titles, authors, and page numbers. It is said that features like the use of bold or italic text, indentation and blank lines help in this recognition. To identify such features and allow text tags to be inserted, the RIDDLE project used the Standardised Generalised Mark-up Language (SGML, ISO 8879). It was reported that the use of SGML allowed the capture of the semantics of the contents pages in a particular catalogue format independent manner; it has provided a platform for the generation of OLC load commands for any OLC format. Semantics of contents page identification was supported by the presence of sufficient consistency between issues of a given journal (that knowledge of the particular journal being processed is all that is required).

Loading the Tagged Output to the Online Catalogue

It is reported that many different catalogues are used throughout Europe. All have some means of loading new information from file. In the RIDDLE project, a mechanism was sought so that a record format for the journal article could fit into a catalogue designed for books. Commercially available tools were used to allow for the translation of the SGML encoded data into a file of load commands for a particular OLC.

The approaches to journal identification that are believed useful are the following.

- Searching for text extraction by OCR software. The text can be a part of the title string or the ISSN number.
- Searching the extracted text for the distinctive pattern of the ISSN number (which is often printed below a bar code as well as appearing on the body of the text).
- Using image recognition to search for an image fragment (e.g., logo).

Tagging the OCR Output

Publishers use different formats to present contents page so that it is possible to clearly identify titles, authors, and page numbers. It is said that features like the use of bold or italic text, indentation and blank lines help in this recognition. To identify such features and allow text tags to be inserted, the RIDDLE project used the Standardised Generalised Mark-up Language (SGML, ISO 8879). It was reported that the use of SGML allowed the capture of the semantics of the contents pages in a particular catalogue format independent manner; it has provided a platform for the generation of OLC load commands for any OLC format. Semantics of contents page identification was supported by the presence of sufficient consistency between issues of a given journal (that knowledge of the particular journal being processed is all that is required).

Loading the Tagged Output to the Online Catalogue

It is reported that many different catalogues are used throughout Europe. All have some means of loading new information from file. In the RIDDLE project, a mechanism was sought so that a record format for the journal article could fit into a catalogue designed for books. Commercially available tools were used to allow for the translation of the SGML encoded data into a file of load commands for a particular OLC.

Overall examination of accessible literature indicates that one of the most aggressively pursued efforts in record-based information generation using OCR (which is critically tailor made towards real application) is that conducted by the United States National Library of Medicine (USNLM). An abstracted profile of this project is described below.

The need for an OCR-based, automated, data entry of bibliographic information from printed journals into an electronic databases seems the main agenda at the National Library of Medicine (NLM), USA (Thoma, 1999). It is claimed that data entry for thousands of bibliographic databases around the world from printed journal articles continues to be mainly manual (CEB, NLM, 1999). At the NLM, however, the data entry for the production of bibliographic fields for MEDLINE (biomedical bibliographic database) is expected to be fully automated through continuous research projects that are underway since the middle of the 1990's; it is a progressive two-phased research project.

As a first step, the R & D division of the library had developed a system called MARS (Medical Article Record System) that involves scanning and converting (by OCR) the author abstracts that appear in the title page of each printed journal article. This system has been in use since 1996 (NLM, 1999).

The second generation MARS system is also developed which automatically extracts the remaining fields (article title, author(s), author affiliation) and the author abstract from the title page of each article in a printed journal (Ford, Hauser, and Thoma, 1999).

It is stated that the MEDLINE database indexes thousands of journals in biomedical sciences

world-wide. It was said that their article title page layouts vary too much and can be categorised into several hundreds. But, all in all three categories were formed. The first group consists of journals having single column, the second group has a combination of single and two columns, and the last group has two columns. It was argued that, it is difficult to design a single automatic labelling module that can handle all types of journals. Therefore, preference is given to group two journals that constitute the majority of the journals indexed in MEDLINE.

This system uses OCR, in addition to program modules that automatically zone the scanned pages, identify the zones as particular fields, and reformat the fields to fit to the MEDLINE conventions (Le et al, 1999). The labelling is based on features calculated from OCR output, and a set of rules derived from an analysis of the page layout for each journal article title page.

The entire process consists of:

- Scan journal images
- Text recognition, which includes detecting zones
- Applying automated labelling which associates a label such as 'Title', 'Author', 'Author Affiliation', and 'Abstract' with each zone of interest.

Text Recognition

Scanned images are sent to OCR software developed by Prime Recognition (PR), Inc. The OCR software first segmented the images into several rectangular text zones. Each zone is then processed to deliver an OCR (recognised text) output including zone co-ordinates, text line information, characters and their bounding boxes, font sizes, and certain style attributes.

The features extracted from the output of OCR system that provide information at the zone, line and character level are given below.

Zone level

- Zone boundaries
- Number of text lines

Line level

- Number of characters
- Average character height
- Average font size

Character level

- Recognised 8-bit character
- Bounding box
- Font size
- Font attributes (normal, bold, underlined, italics, superscript, subscript, etc.)

Zone Features for Document Labelling

The geometric and non-geometric features of text zones are considered for document labelling. The geometric layout features are calculated based on the zone location, zone order, and zone dimension. For instance, it was reported that title zone is usually located on the top half of the first page of an article with the biggest font size. Le et al (1999) shows, citing J.H. Ling-who conducted a title page study, that about 96% of titles have the largest font compared to other zones in the top half of the first page. To further attribute the geometric features of zones, author and affiliation follow title zone. In addition, the font size of the author zone is usually smaller than that of the title zone.

The non-geometric features are derived from the zone contents. Most commonly used non-geometric features include: total characters, total capital letters, font size, total punctuation marks like commas, semi-colons, etc.; number of words, number of degrees (MD, Ph.D.,...), existence of key terms (abstract, summary; aim, background, design, result, ...); key word; introduction, received, revised); etc.

In order to classify each zone into meaningful fields, two systems were put into use: rule-based and neural network systems.

Labelling Using Rule-Based System

As an example to the application of rules for field classification, to label a zone as 'Title' the following conditions are checked.

1. Zone's font size should be maximum.
2. Number of degrees should be less than three or percentage of degrees less than ten
3. The value of the zone's upper co-ordinate should be less than height of article divided by three.
4. The value of the zone's lower co-ordinate should be less than height of article divided by two.

After incorporating the above features and other details to a rule-based system, an experiment was conducted on 38 journals consisting of 1407 articles. Out of these articles, 1402 were labelled correctly with 99.6% accuracy. The five errors were attributed to in labelling affiliation zones due to the incorrect font attributes and poor contents obtained from the output of the PR OCR system.

Labelling Using Neural Network System

16 different journals consisting of 66 issues were selected for the experiment for a total of 2176 images. The Back-Propagation (BP) neural network was put into use. This network was implemented with an input layer (38 text zone features), 5 output layer (title, author, affiliation, abstract and others) and one single hidden layer of which the number of nodes is 16. The BP neural network was trained with the test data. It was reported that the average classification accuracy on the test data set was about 97.0%. Most errors were due to the segmentation problem generated from the OCR output that split a given zone (such as title zone) into multiple zones, as well as merged several different zones such as author and affiliation zones) into a single zone.

CHAPTER THREE: JOURNALS PUBLISHED IN ETHIOPIA

3.1. Introduction

Modern higher education in Ethiopia began in the middle of the 20th century. This was with the founding of the University College of Addis Ababa in 1950 (Engida, 1999). Soon after this period, several colleges and faculties were established and became under the umbrella of the University College of Addis Ababa (which was upgraded to a university under the name of Haileselassie I University, currently the Addis Ababa University). These include the Faculty of Science, the Faculty of Arts, the Faculty of Education, the School of Social Work, the College of Business Administration, the Law School and the Faculty of Medicine. The 1970s and 1980s saw the emergence of new institutions of higher learning in the country, and more have come into being since.

The expansion of modern higher education paved the way for the formation of different scientific societies in various faculties and colleges. Apart from conducting lectures, the academic community began conducting research to empower education and bring about socio-economic development. With respect to Addis Ababa University, this effort was backed up (about 40 years ago) by a Charter of Haile-sellassie I University that stated research and scholarship as one of the major components of the duties of the University (Endashaw, 1995). This demanded the production of scientific journals where research outputs could be reported.

Evidence (RPO, 1994) shows that the oldest professional journal in Ethiopia is the *Ethiopian Medical Journal* which is still published. The journal first appeared in 1962. After this period, several journals in different disciplines have emerged: *Journal of Ethiopian Studies*

(1963), *Journal of Ethiopian Law* (1964), *ZEDE: Journal of Ethiopian Association of Engineers and Architects* (1965), *Ethiopian Journal of Education* (1967), etc.

At present, there are about 30 journals in Ethiopia produced by scientific societies, associations, and academic and research institutions. However, it should be realised that this figure shows only journal titles that are received by the Legal Deposit and National Bibliography Team (LDANBT) of the National Library.

The table in Appendix 1 shows the details of each journal, beginning with the latest issue received by LDANBT, back for five consecutive volumes. It is the result of the physical count of the journals made by the researcher.

Column cells showing question mark (?) refer to missing journal issues, i.e., journal issues which might have been published but are not received by LDANBT. Columns marked with "x" refer to issues that are not published at all. Thus, the "x" sign is placed for newly emerging journals, the required volumes of which have not been published.

It shall be noted that the number of articles, communications, case reports included in each cell (to show details) are grouped together and are considered as articles, in the total number of articles column of the table. The reasons for merging the different attributes as articles is the result of the fact that the order of presentation of field for articles, communications and case reports is the same in a given journal issue.

The number of articles listed for bilingual journals refers to articles in the English language. Bilingual here refers to articles published in English and Amharic.

3.2.2. Analysis of Characteristics among Journals

An examination of journals indicates both similarities and differences in the number of fields and their layouts for article contributions among journal titles.

Regarding their number, there are variations to the extent that the article title pages of some journals contain only article title and author(s), and some others have all bibliographic elements (journal title, volume and issue numbers, year, page range, article title, author(s) and abstract). For instance, the *Journal of Ethiopian Law, Ethiopian Journal of Languages and Literature*, and that of *WALIA: Journal of the Ethiopian Wildlife and Natural History Society* are good examples of the former. On the other hand, *SINET: Ethiopian Journal of Science*, *Pest Management Journal of Ethiopia, Ethiopian Journal of Economics* and *Bulletin of the Chemical Society of Ethiopia* are typical examples of the latter.

The rest of the journals are in between these two extreme cases, and the number and layouts of their bibliographic fields vary considerably. However, for journals having the abstract attribute, the order of presentation of article title, author(s) and abstract are consistent and in that order from top to bottom in an article's title page. Appendix 2 shows lists of journals based on the number and order of presentation of article title page bibliographic fields.

3.3. Classification of Journals

3.3.1. Introduction

The classification of journals can be made under different levels of article title page layout similarity among journals. The attributes that can enable the grouping of journals into different strata are discussed below.

3.3.2. Abstract as an Attribute of Classification

The author abstract is an important element of classification. Overall, journals are grouped into two based on whether journal articles include an author abstract or not.

Table 3.1. shows classification of journals based on the absence/presence of the author abstract.

Table 3.1. Classification of Journals based on the absence/presence of author abstract

Journals with author abstract	Journals with no author abstract
1. Journal of the Ethiopian Society of Chemical Engineers 2. Ethiopian Journal of Health Sciences 3. Educational Journal 4. Ethiopian Journal of Health Development 5.Journal of the Ethiopian Veterinary Association 6.Ethiopian Pharmaceutical Journal 7.Water : Ethiopian Journal of Water Science and Technology 8.Ethiopian Medical Journal 9.Ethiopian Journal of Agricultural Economics 10. Ethiopian Journal of Development Research 11. The Ethiopian Journal of Education 12. Eastern Africa Social Science Research Review 13. Journal of Ethiopian Medical Practice 14. Journal of the Ethiopian Society of Mechanical Engineers (ESME Journal) 15. SINET : Ethiopian Journal of Science 16. Ethiopian Journal of Agricultural Sciences 17. JESA (Journal of Ethiopian Statistical Association) 18. Ethiopian Journal of Economics 19. Bulletin of the Chemical Society of Ethiopia 20. Pest Management Journal of Ethiopia 21. ZEDE: Journal of Ethiopian Architects and Engineers Association	1. Ethiopian Journal of Languages & Literature 2. Journal of Ethiopian Law 3. HISSAB : An Ethiopian Journal of Mathematics 4. Ethiopian Association of Civil Engineers Bulletin 5. WALIA : Journal of the Ethiopian Wildlife and Natural History Society 6. Association of Ethiopian Architects Journal 7. IER Flambeaw 8. Journal of Ethiopian Studies 9. Research Bulletin of the Gondar College of Medical Sciences

Journals listed under the second column ("Journals with no author abstract") of the above table are excluded from further consideration. This is because such journals can be considered for another study directed towards capturing bibliographic data from the tables of contents of journal issues. The justification behind this idea is that, as long as there is no the author abstract, scanning and OCR each article title page of journal issues is tedious and demands much more time. While it is possible to capture such bibliographic elements as journal title, volume and issue numbers, year, article title, author(s) and first page location of a journal issue at once from the table of contents, it is not necessary to scan each article title page to get the same result. Hence, subsequent discussions concentrate on journals having an author abstract and one or more of the rest of the bibliographic elements in their article title page.

3.3.3. Absence / Presence of Journal Title, Volume, Issue Number, Year and Page Range as Attributes of Classification

Journals none of which contain none of such fields as journal title, volume and issue numbers, year and page range are listed below (in fact there are only two journals that do not have the attributes mentioned). Regarding these journals, containing only the article title, author(s) and abstract, it is not necessary to include them for experimental purpose. This is because the order of presentation of these fields for all journals containing the author abstract is consistent and the same. Therefore, an algorithm designed for journals containing these and other fields (journal title, volume and issue numbers, year and page range) is sufficient enough, as the former set of journals are the subset of the later category. Hence, few amendments made to the algorithm designed for the later group of journals can capture the article title, author(s) and abstract fields of journals with out any other attributes. The only amendment needed is to exclude program modules designed to capture the journal title,

volume and issue numbers, year and page range for journals with full set of bibliographic fields. In fact there must be manual intervention in order to encode these fields in order to build a full bibliographic record of a journal article.

Table 3.2: Journals with article title, author(s) and abstract as their only attributes

1	Educational Journal
2.	Journal of the Ethiopian Society of Chemical Engineers

3.3.4. Location of Journal title, Volume, Issue Number, Year and Page Range as an Attribute of Classification

It should be realised that the discussion is about journals with article title, author(s) and abstract and one or more of Journal title, volume, issue number, year and page range as their attributes. Whether one or more of Journal title, volume, issue number, year and page range are located at the top most, bottom most or back of article title page matters in how to create a program capable of capturing these attributes. Due to this fact, journals are classified based on the location of these fields on an article title page or its back page. Table 3.3. shows journals grouped based on the location of the fields Journal title, volume, issue number, year and page range.

Table 3.3: Classification of journals based on the location of the fields Journal title, volume, issue number, year and page range

Journals in which one or more of the fields journal title, volume, issue number., year and page range are at the:		
Top of article title page	Bottom of article title page	Back of article title page
1. Ethiopian Pharmaceutical Journal 2. Ethiopian Medical Journal 3. Ethiopian Journal of Development Research 4. The Ethiopian Journal of Education 5. SINET : Ethiopian Journal of Science 6. Ethiopian Journal of Agricultural Sciences 7.JESA (Journal of Ethiopian Statistical Association) 8. Ethiopian Journal of Economics 9.Bulletin of the Chemical Society of Ethiopia	1.Journal of the Ethiopian Veterinary Association 2.Water : Ethiopian Journal of Water Science and Technology 3.Journal of Ethiopian Medical Practice 4. Pest Management Journal of Ethiopia 5.ZEDE: Journal of Ethiopian Architects and Engineers Association	1.Ethiopian Journal of Health Development 2.Ethiopian Journal of Agricultural Economics 3.Eastern Africa Social Science Research Review 4.Journal of the Ethiopian Society of Mechanical Engineers (ESME Journal) 5.Ethiopian Journal of Health Sciences

Journals listed in column three of Table 3.3 are excluded from further consideration. The main reason is that it is uneconomical in terms of time and labour to scan the back of article title pages for the sake of capturing only one or more of Journal title, volume, issue number, year and page range. Hence, the same arguments given in section 3.3.3 above apply in dealing with the rest of the fields (article title, author(s) and abstract) located on an article's title page of these group of journals.

The journals in columns one and two contain the required bibliographic fields in their article title page, though the location of one or more of the fields Journal title, volume, issue number, year and page range varies. Even though these journals constitute the possible candidates for experimentation, it is not possible to develop a program that can handle both

groups of journals within the available time limit for the research project. As a result, it became mandatory to choose one group of these journals and develop the prototype program for them. Accordingly, the first group (journals with one or more of the fields Journal title, volume, issue number, year and page range are at the top of article title page) are selected, since they constitute the majority (9 (64.3%) journals) as compared to group two journals, which are 5 (35.7%) in number.

4.1. Introduction

In this chapter strategies followed for field segmentation and classification are discussed and tested. This is based on the literature reviewed in Chapter 2 and the layout features of journal article title pages considered in Chapter 3. Moreover, the output of the algorithm is formatted into the structure of ISO-2709 format so that the text retrieval system (*CDS/ISIS for Windows*) used at NALA can understand the data.

The most widely used algorithms in document analysis and document understanding consider the geometric and non-geometric features of documents. The geometric features widely referred to include zone location, zone order and zone dimension. The non-geometric once include number of characters and text-lines in a zone; font style and size; punctuation marks like comma (','), full stop ('.'), braces ('(',')'), etc.; key terms; character height and width; spacing between consecutive text-lines; etc.

For the present research, effort has been made to develop algorithms that incorporate some of these geometric and non-geometric features that are relevant in the segmentation (document analysis) and classification (document understanding) of fields for journal articles published in Ethiopia. The results for the application of the field segmentation and classification algorithms described below are presented later in this chapter.

The major components (processes) of the system developed known as *Ethiopian Articles Recording System (EARS)* are given in a diagram (see Figure 4.1), including the formatting process of *EARS* output to ISO 2709 format (which is compatible with *CDS/ISIS for*

Windows).

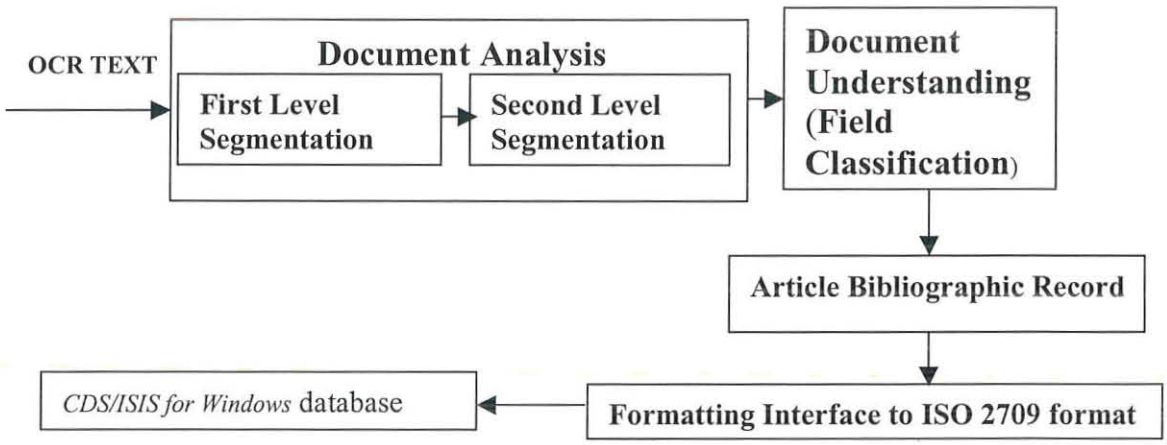


Figure 4.1. Diagrammatic Representation of Major *EARS* Components (processes)

4.2. Journals for Experimentation

As already shown in section 3.3.4 of Chapter 3, the journals selected for the experiment are nine in number. They are the journals in which one or more of the fields such as journal title, volume, issue number, year and page range are located at the top of a given article title page. Experimentation is carried out on two articles (one from the last issue received by LDANBT and the other from the fifth issue from the last) of these journals. Totally, from nine journals, 18 articles are used for experimentation. This has the advantage of ensuring that the program handles journal layouts that vary over time. However, for the journals examined, there is no any significant layout variation over time (five consecutive volumes). In fact, there are minor changes in layout like the *Ethiopian Medical Journal* shown in Figure 3.1, Chapter 3. In that figure, the 1995 issue shows that the journal title is followed by volume without any punctuation mark; and at the end of the volume, there exists a comma. On the other hand, the 1999 issue shows that journal title and volume are separated by comma, and volume is followed by full stop. Even these kind of minor layout variations are rarely found in most journals and these variations are not hard to programme and are easily handled.

As the result of the experiment, two articles (one from the 1992 issue of the *Ethiopian Journal of Agricultural Sciences*—written as *Ethiop.J.Agric.Sci.* in article title page, and the other from the 1992 issue of the *Ethiopian Journal of Economics)* cannot be segmented and classified properly. This is because the OCR software interpreted the key term "ECONOMICS" as "ECONOMIC," and the key term "Agric" as "@". Therefore the second level segmentation algorithm can not find journal identifier key words for successful segmentation and the consequent activity of field classification. These problems can only be managed by applying OCR software with better accuracy rate.

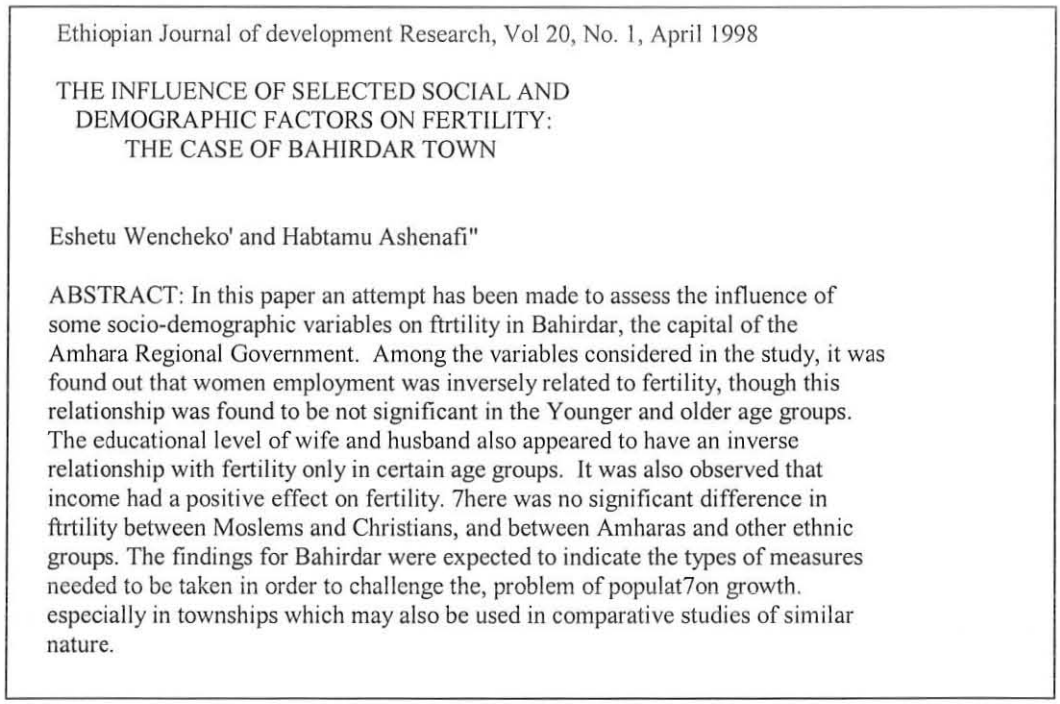
4.3. Field Segmentation

A two-level approach is followed for the segmentation of fields available in the article title page of journals. The first level segmentation produces four text zones consisting of one or more of journal title, volume, issue number, year and page range as text zone one, article title as text zone two, author(s) as text zone three and author abstract as text zone four. The second level segmentation considers text zone one and degenerate it into its constituent parts such as journal title, volume, issue number, year and page range. The details of the algorithms used for both levels of segmentation are given below.

4.3.1. First Level Segmentation

The layout analysis of journal article title pages shows that the bibliographic elements form four major text zones which are separated from each other by one or more white line (s); while white line between text-lines in each zone is consistent and narrower, there is a wider line spacing that commences the end of a given text zone. The concept of white line spacing as text zone extraction mechanism has been used by Tsujimoto and Asada (1992), Liang et al

(1996) and that of Weibel, Oskins and Vizine-Goetz (1989) as described in Chapter 2. Tsujmoto and Asada (1992) have combined adjacent text lines from an input text file into a block as long as the text lines have the same typeface and are consistent in line spacing. Liang et al (1996) realised that when there is a significant change in text line heights, inter-text line spacing, or justification in the input image, it means that a new text block has begun. In the same way, Weibel, Oskins and Vizine-Goetze(1989) have used the concept of line spacing in order to form what they called compound tokens (functionally meaningful units that can be evaluated for their relevance to bibliographic field segmentation). Thus, compound tokens are formed by adding tokens (words) to a compound structure until either font style changes, font size changes or vertical white space exceeds a certain limit. For the present research, the concept of font size and font style can't be useful. This is because the OCR software produced an ASCII text regardless of font size and style variations in the original document. Therefore, only the concept of white line spacing is used in the segmentation of text zones for the first level segmentation process. In fact, white line spacing is reasonably sufficient enough to meet the first level segmentation requirements of journal articles. This concept can easily be illustrated by showing a sample article title page.



Ethiopian Journal of development Research, Vol 20, No. 1, April 1998

THE INFLUENCE OF SELECTED SOCIAL AND DEMOGRAPHIC FACTORS ON FERTILITY: THE CASE OF BAHIRDAR TOWN

Eshetu Wencheko' and Habtamu Ashenafi"

ABSTRACT: In this paper an attempt has been made to assess the influence of some socio-demographic variables on ftrtility in Bahirdar, the capital of the Amhara Regional Government. Among the variables considered in the study, it was found out that women employment was inversely related to fertility, though this relationship was found to be not significant in the Younger and older age groups. The educational level of wife and husband also appeared to have an inverse relationship with fertility only in certain age groups. It was also observed that income had a positive effect on fertility. 7here was no significant difference in ftrtility between Moslems and Christians, and between Amharas and other ethnic groups. The findings for Bahirdar were expected to indicate the types of measures needed to be taken in order to challenge the, problem of populat7on growth. especially in townships which may also be used in comparative studies of similar nature.

Figure 4.2. An OCR result article title page from the *Ethiopian Journal of Development Research*

From Figure 4.2, it is easier to realise that there are four text zones separated by white line(s). From top to bottom, the first zone consists of such bibliographic fields as journal title, volume, issue number and year (this journal doesn't have page range-where the article appears in a journal issue). The second zone is the article title. The third and fourth zones are the author(s) and author abstract, respectively.

Even though, the number of fields in the first zone varies from journal to journal, the order of presentation of these four zones is consistent and in that order for the selected journals. Therefore, to segment an input OCR text (like the one shown in Figure 4.2. above) into these four zones, an algorithm that incorporates line spacing as an indication that commences the end of a given text zone is utilised. Technically, white line spacing has been obtained if two consecutive characters read in from an input file are line-feed characters. The flowchart for

this procedure for a text zone is given in Figure 4.3.

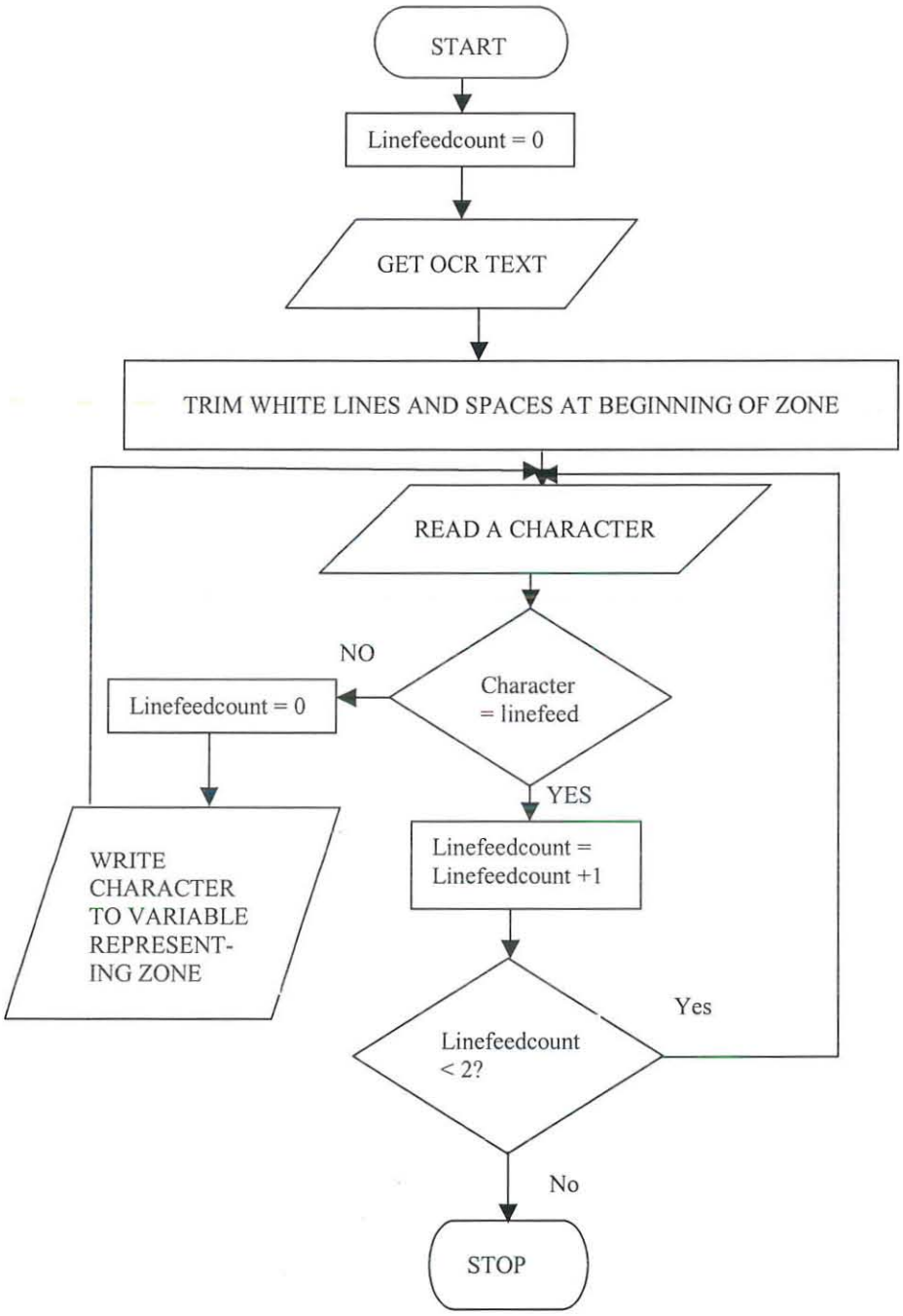


Figure 4.3. Flowchart for the first level segmentation algorithm

In order to handle the result of the first level segmentation process, four temporary string variables have been defined. Each variable is then filled with text characters read in from an input file until a white line is detected. In order to consider these zones for classification and

then send them to an output file, a second level segmentation has been made that partitions the fields of the first text zone into its component parts.

4.3.2. Second Level Segmentation

The last three text zones (article title, author and abstract) formed by the first level segmentation algorithm remain intact and are considered for field classification and labelling without any further field segmentation process. However, the first text zone should be considered for further segmentation into its building blocks. The problem, in this regard, however, is the possibility to use neither inter-word nor inter-line spacing. Furthermore, there is no any font style and size, character height or width changes among these fields. Hence, the only possible outlets observed that can meet the segmentation requirements of the first text zone are certain punctuation marks like comma, colon, braces, end of line mark and counting the number of words that form a given field in cases there are no any punctuation marks to segment the field. In his research project, Kebede (1997) has utilised various punctuation marks of the Anglo-American Cataloguing Rules 2 in order to segment and classify OCR text for book catalogues. These include full stop, colon, full stop-space-dash, comma, semi-colon, etc. For the present research, various punctuation marks become the corner stone for the segmentation of different fields in text zone one. For instance, given text zone one, a journal title might be separated from volume by comma; volume and issue number might be separated either by comma or brace ("()"). Similarly, page range might be separated from any other field by comma, colon, space or semi-colon, etc.

The other challenge is the fact that the order of presentation and number of fields in the first zone are journal specific. In other words, while the number of fields is one in certain journals it is three, four or five in others. For instance, only the journal title forms the first zone for the

Ethiopian Pharmaceutical Journal. On the other hand, the first text zone of the *Ethiopian Journal of Development Research* (Figure 4.2.) consists of four fields and that of the *Bulletin of the Chemical Society of Ethiopia* and *Ethiopian Journal of Economics* is five.

Due to strong attachments between the number and order of fields (in the first text zone) and a journal title, it was a must to have some mechanism of identifying a given journal in order to have a functional second level segmentation algorithm. To manage layout variations, Harrison, Roos and Thomas (1995) have incorporated journal specific knowledge to their system developed to automatically extract article information from the table of contents of journal issues and load it to an online catalogue. Journal identifier information has been extracted from text available in journal issues content page that include ISSN, a string obtained from the journal title, bar code and logo. For the present research, ISSN, barcode and logo become irrelevant. This is because not all article title pages of journals selected for the experiment possess one or more of these attributes. Hence a string extracted from the title of journals is considered. Thus, extracting words from title of journals as they appear in the first page of a journal article to form a dictionary of keywords is utilised. Some of the journal titles are abbreviated or have acronyms while others have the full form of the journal title. The list of journals selected for the experiment and the keyword considered to build the dictionary are shown in Table 4.1. The title of journals is the one that appears in their article title page.

Table 4.1. Titles of journals (as they appear in the article title page of journals) and the keyword considered for developing a dictionary

No.	Journal title	keyword
1.	Ethiopian Journal of Development Research	Development
2.	Ethiop Med J	Med
3.	Ethiop. J. Agric. Sci.	Agric
4.	Eth. Pharm. J.	Pharm
5.	Ethiopian Journal of Economics	Economics
6.	The Ethiopian Journal of Education	Education
7.	JESA	JESA
8.	SINET: Ethioip. J. Sci.	SINET
9.	Bull. Chem. Soc. Ethiop	Chem

The selection of keywords is not random. A word from the title of a journal is selected for the dictionary provided that it represents the content of the journal or if it is a unique abbreviation. For example, the terms 'Development', 'Med', etc. are selected because they represent the content of the respective journals and can uniquely identify them from any other journal. The acronym JESA is also important in order to detect the *Ethiopian Journal of Statistical Association* from any other journal title.

Technically, these keywords are kept in a file and are loaded into memory using a binary tree whenever there is a need to search for the identity of a journal to process an input text. Regarding the procedures of operation, the dictionary terms are loaded into memory. Then a word is extracted from the first text zone and kept in a variable. The value of this variable is then searched in the database using a binary search technique for its availability. If available, it means the term represents identification for a journal and hence different functions that can segment the components of the first text zone are triggered (the journal specific functions

incorporate different punctuation marks that are specific to that journal). As a result, different fields are formed from the first text zone that completes the field segmentation process for a given journal. The following algorithm (Figure 4.4) shows the operations for a specific journal (*Ethiopian Journal of Development Research*) such as that shown in Figure 4.2. Note that, for each journal a specific feature, based on functions, is developed to undertake second level segmentation.

```
BEGIN
WHILE (NOT END OF FIRST TEXT ZONE)
        REPEAT
                READ a character and WRITE it to a temporary variable (that
                represents the journal title)
        UNTIL comma (,); ignore non-alphabetic characters
        REPEAT
                READ a character and WRITE it to a temporary variable (that
                represents volume)
        UNTIL comma (,); ignore non-alphabetic characters
        REPEAT
                READ a character and WRITE it to a temporary variable (that
                represents Issue number)
        UNTIL comma (,); ignore non-alphabetic characters
        REPEAT
                READ a character and WRITE it to a temporary variable (that
                represents Year)
        UNTIL new line
END.
```

Figure 4.4. Second level segmentation algorithm for the *Ethiopian Journal of Development Research*

On the other hand, if the space-delimited string (word) extracted from the first text zone is not in the dictionary of terms, another term extraction and comparison process is conducted recursively through a recursive function either until the key word is obtained or the space-delimited strings are finished from the first text zone. The flowchart representing the second level segmentation algorithm is depicted in Figure 4.5 next page.

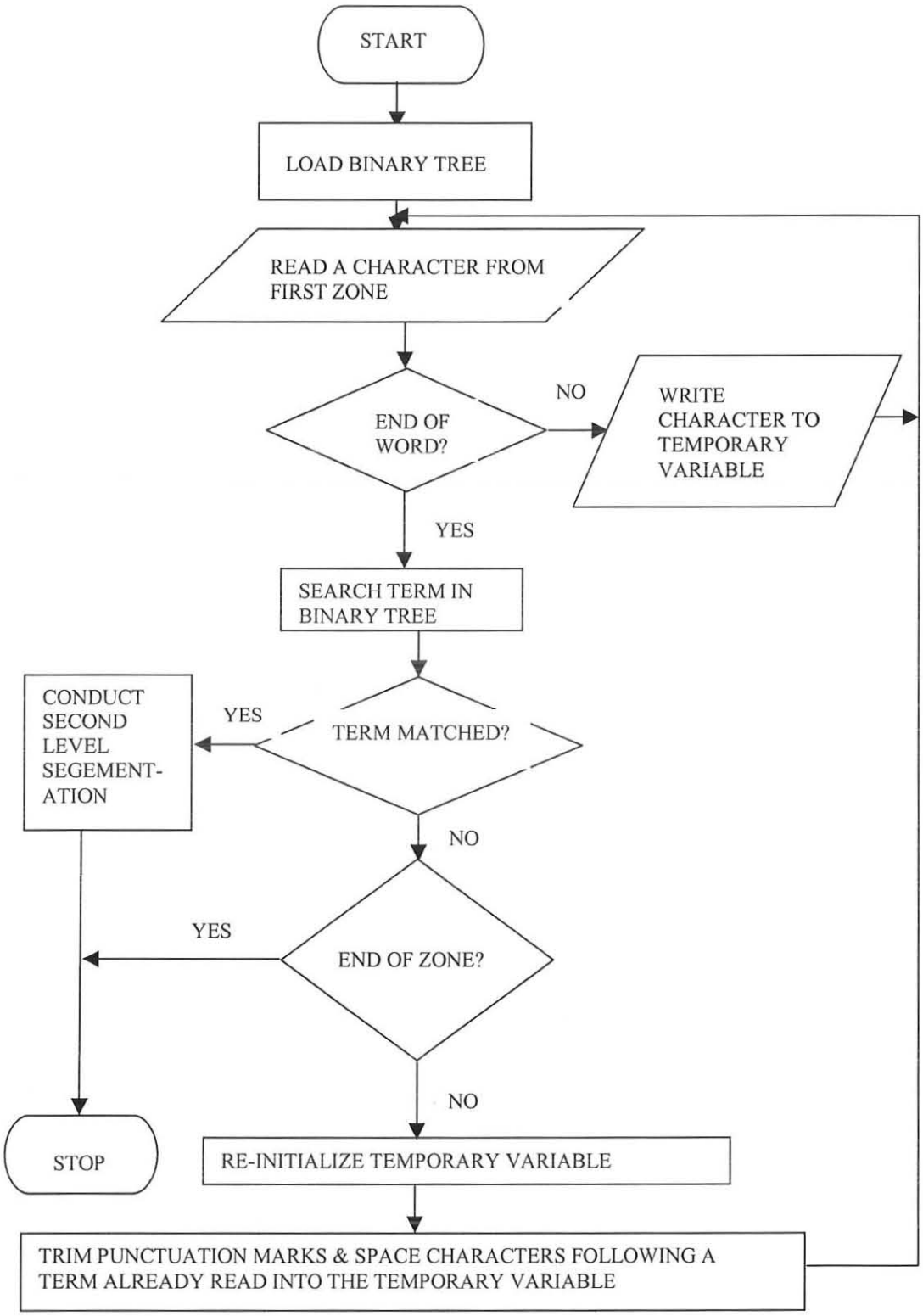


Figure 4.5. Flowchart for the second level segmentation algorithm

4.4. Field Classification

The aim of the whole subject of document analysis and document understanding is to get searchable blocks of text, so that the text can be manipulated by a Database Management System (DBMS) or Information Retrieval System (IRS).

During the document analysis (field segmentation) phase (discussed so far), various coherent components of an article's title page text are partitioned and kept in temporary variables. However, they are not yet formally identified as to the content of which variable represents journal title, volume, issue number, year, page range, article title, author (s) and abstract. To do so, knowledge about the features of the different fields should be possessed. For instance, the order of zones in an article title page is a very important mechanism of understanding document components. This concept has been used by Le et al (1999) in the labelling of article title, author (s), author affiliation and author abstract for indexing journal articles. For example, the article title precedes the author and author affiliation zones. Tsujimoto and Asada (1992) also used the same concept in an effort to develop a complete text reading system. They reported that document layout is designed according to the human reading manner; and human reading manner assumes that, given a document, the title precedes the abstract, chapters and sections, while subtitles precede paragraphs.

In the present research, as already stated in Chapter 3, article title, author(s) and author abstract appear in that order from top to bottom in an article's title page for the journals examined. This knowledge was important in assigning the three fields mentioned into their respective meanings: Article title, Author(s) and Abstract. The knowledge about zone order was incorporated in the segmentation algorithm by assigning sequential number to the temporary variable that holds a given zone.

Classification of fields in the first text zone is handled during the second level field segmentation stage. Whenever a journal's identifier key word is detected from the input text file, the functions that manipulate the first text zone of that specific journal identify a given field as one of Journal title, Volume, Issue Number, Year and Page Range by giving a serial number to the temporary variables that hold the contents of the segmented fields.

Finally, all the values of the temporary variables are transferred to a class that contains the various fields of a journal article. The contents of the class are then written to an output file. The content of the output file, in turn, is formatted into an ISO 2709 data structure so that it can be manipulated by *CDS/ISIS for Windows* (text retrieval software). This software is selected because currently, *CDS/ISIS for Windows* is used to automate different databases of NALA. The description on how the output data of the program developed known as *Ethiopian Articles Recording System (EARS)* is formatted into the data structure of ISO 2709 is given later in this chapter.

4.5. Programme Testing

4.5.1 Test Cases

From the five consecutive volumes of each journal selected for the experiment (nine in number), the articles (97 in number) of two issues (one from the last and the other the fifth from the last volumes received by the LDANBT) are considered as test cases. This is excluding the 18 articles that are used for experimentation.

Regarding the acquisition of the issues for test cases, the two issues for each journal are collected from different sources particularly from libraries and journal publishers. This is because the rules of the LDANBT do not permit the withdrawal of any journal issue out of

their storage room. Neither the Team has a scanner to get the image of the articles required.

4.5.2. Text Recognition

Word Scan 3.0 OCR software is used in order to produce machine-readable text from the article title page of journals selected for the experiment.

An article title page is scanned using the "ACQUIRE IMAGE" option of *Word Scan 3.0.* The image is brought to the screen with a number of text zones automatically created by the OCR software. The automatic zoning facility of *Word Scan 3.0*, however, became irrelevant. This was because the software was inconsistent in that it created several text zones out of text that should have belonged to the same text zone. For example, it either creates two text zones out of an article's title or merges two zones such as the first text zone and the article title or article title and author (s).

Consequently, the auto zone feature of the OCR software is ignored. Rather, part of the article title page text from the top (consisting of the first text zone) to the author abstract are selected with the mouse for subsequent recognition. This leaves out any text following the author abstract. The resulting image is then given to the OCR module of *Word Scan 3.0* by choosing the "OCR" option from the menu. This generates machine-readable text of the image. Finally, the output of OCR is then saved on a disk for subsequent processing by the automatic segmentation and labelling programme, i.e., *EARS.*

4.5.3. Analysis of Results

The results of *EARS* for the test cases are shown in Table 4.2. below.

Table 4.2: Program Test Results

No.	Journal Title	Recognised		Mis-recognised		Total records
		No.	%	No.	%	
1.	Ethiopian Journal of Development Research	3	75	1	25	4
2.	Ethiopian Medical Journal	10	90.91	1	9.09	11
3.	Ethiopian Journal of Agricultural Sciences	12	70.59	5	29.41	17
4.	Ethiopian Pharmaceutical Journal	12	100	-	-	12
5.	Ethiopian Journal of Economics	4	66.67	2	33.33	6
6.	The Ethiopian Journal of Education	5	100	-	-	5
7.	Journal of Ethiopian Statistical Association	7	87.50	1	12.50	8
8.	SINET: Ethiopian Journal of Science	14	77.78	4	22.22	18
9.	Bulletin of the Chemical Society of Ethiopia	16	100	-	-	16
TOTAL		83	85.57	14	14.43	97

As Table 4.2 shows, out of 97 articles given to the system, 83 (85.57%) are correctly segmented and classified. The remaining 14 (14.43%) records are not correctly segmented and labelled, because of two main reasons. Firstly, the OCR software created mis-recognised (or rejected) characters in a candidate keyword in the journal title and the various punctuation marks that mark the beginning or end of a field in the first text zone. Mis-recognition of characters mostly occurs when the original document is not clear. Secondly, article titles with larger fonts in the original document are disturbed by the OCR software, since it introduces white line between text lines (that doesn't exist in the original printed document) while forming the article title zone. This later problem is mostly the result of article titles with large font size.

Both problems had bad consequences for field segmentation and classification algorithms. Mis-recognised characters create problems of key term identification in the second level segmentation phase. For instance, when a key term is mis-recognised, term matching with dictionary of terms in the database never happen. Moreover, when various punctuation marks with in the fields of the first zone are mis-recognised or rejected, they create confusion for the second level segmentation algorithm. Thus, if journal title and volume of a given journal are separated by comma in the original document but the comma is rejected or mis-recognised during OCR, then the algorithm merges volume with journal title assuming that it is part of the journal title.

On the other hand, introducing white lines that does not exist in the original document affects the first level segmentation algorithm. It is said that the first level segmentation algorithm depends on the presence of one or more white lines as a mechanism for separating different text zones. Hence, unexpected white line with in the same text zone becomes a bottleneck for the algorithm.

Even though both problems affect the field segmentation algorithms, they have also a negative impact on the field classification algorithm. This is because the correct classifications of fields depend entirely on their correct segmentation.

The problem of introducing white line in the article title zone (when the font size of the article title is large) is considered as the problem of the algorithm used. Consequently, the error rates are classified in to two. The first error (error1) is the result of mis-recognition of characters (by the OCR software) forming the keyword used for building the dictionary and

the various punctuation marks that serve to partition the fields in the first text zone. This is entirely considered the problem of the OCR software. The second error (error2) is the one that occurred when article title in the original document is in large font but when the OCR software introduces white line in between lines of text with in the title zone. Even if the algorithm used does not tolerate this problem, it is a problem that might be solved if a different algorithm that can expect white line between text lines in the article title zone is used. Therefore, the second error (error2) is considered as the problem of the present algorithm. From the total records (14 in number) that are mis-classified, 12 are the result of the first error type and the remaining 2 (from the *Ethiopian Journal of Agricultural Sciences*) are attributed to type two error. The distribution of these errors is shown in the following table.

Table 4.3: Share of error types with respect to the OCR Software and that of the algorithm used

Total Mis-recognised records	error1		Error2	
	No	%	No.	%
14	12	85.71%	2	14.29%

4.6. Formatting the Output of EARS into ISO 2709 Format

The different databases that are available at the Technique and Computer Services of NALA are developed using *CDS/ISIS for Windows*. This text retrieval system can only understand data if it is available in ISO 2709 format. Hence, in order to manipulate the output data of EARS by *CDS/ISIS for Windows*, the data should be formatted to a data structure that conforms to that of ISO 2709 format. In the following few sections, the descriptions of procedures and techniques used for the same purpose are given.

the value for the inquiry, "Number of fields:" in the second line of block 1 in Figure 4.6. The first line of block 1 is simple description, and hence, not evaluated as to what it contains by Fangorn. Line 5 of block 1 is given a value of hash /#/ which is user defined. This is a very important symbol used to define the boundary between records in an input file. This symbol should be put in a line in the input file with no other character in that line. Lines 3 and 4 can be filled with any record delimiter symbol. But, if one of the three lines (lines 3, 4 or 5) is chosen, it is not mandatory to have values for the rest.

Block 2 is entirely left for field delimiters within a record. As an example, the tag used for journal title, "jou:" in the input file is shown in the above specification file. The tag assigned for this field in the ISO-2709 file is chosen (randomly) to be 005. Block 2 should be repeated 8 times to specify input and output field tags for all fields. The tags used to structure the input data (the one produced by EARS) and its equivalent in the ISO 2709 format is shown in Table 4. 4.

Table 4.4. Input and ISO 2709 file tags used for EARS output data

No.	Field tag in the input file	Field tag in ISO 2709 file	Description
1.	jou:	005	Journal title
2.	vol:	010	Volume
3.	num:	015	Issue number
4.	yea:	020	Year
5.	pag:	025	Page range
6.	art:	030	Article title
7.	aut:	035	Author
8.	abs:	040	Abstract

Note: both tags in columns 1 and 2 of Table 4.3 are user defined.

the value for the inquiry, "Number of fields:" in the second line of block 1 in Figure 4.6. The first line of block 1 is simple description, and hence, not evaluated as to what it contains by Fangorn. Line 5 of block 1 is given a value of hash /#/ which is user defined. This is a very important symbol used to define the boundary between records in an input file. This symbol should be put in a line in the input file with no other character in that line. Lines 3 and 4 can be filled with any record delimiter symbol. But, if one of the three lines (lines 3, 4 or 5) is chosen, it is not mandatory to have values for the rest.

Block 2 is entirely left for field delimiters within a record. As an example, the tag used for journal title, "jou:" in the input file is shown in the above specification file. The tag assigned for this field in the ISO-2709 file is chosen (randomly) to be 005. Block 2 should be repeated 8 times to specify input and output field tags for all fields. The tags used to structure the input data (the one produced by EARS) and its equivalent in the ISO 2709 format is shown in Table 4. 4.

Table 4.4. Input and ISO 2709 file tags used for EARS output data

No.	Field tag in the input file	Field tag in ISO 2709 file	Description
1.	jou:	005	Journal title
2.	vol:	010	Volume
3.	num:	015	Issue number
4.	yea:	020	Year
5.	pag:	025	Page range
6.	art:	030	Article title
7.	aut:	035	Author
8.	abs:	040	Abstract

Note: both tags in columns 1 and 2 of Table 4.3 are user defined.

The appearance of an article (such as that shown in Figure 4.2. for the *Ethiopian Journal of Development Research*) with the field tags automatically incorporated in the EARS output file is illustrated in Figure 4.7.

```
jou: Ethiopian Journal of development Research
vol: 20
num: 1
yea: April 1998
pag:
art: THE INFLUENCE OF SELECTED SOCIAL AND
     DEMOGRAPHIC FACTORS ON FERTILITY:
          THE CASE OF BAHIRDAR TOWN


aut: Eshetu Wencheko' and Habtamu Ashenafi"


abs: In this paper an attempt has been made to assess the influence of
   some socio-demographic variables on ftrtility in Bahirdar, the capital of the
   Amhara Regional Government Among the variables considered in the study, it was
   found out that women employment was inversely related to fertility, though this
   relationship was found to be not significant in the Younger and older age groups.
   ne educational level of wife and husband also appeared to have an inverse
   relationship with fertility only in certain age groups. It was also observed that
   income had a positive effect on fertility.7nere was no significant difference in
   fertility between Moslems and Christians, and between Amharas and other ethnic
   groups. The findings for Bahirdar were expected to indicate the types of measures
   needed to be taken in order to challenge the, problem of populat7on growth.
   especially in townships which may also be used in comparative studies of similar
   nature.

#
```

Figure 4.7. Structure of a record (from the *Ethiopian Journal of Development Research*) in the EARS output file

After Fangorn has been given the name of the *Specification File*, with the required details filled out properly, and the input text file is structured as that shown in Figure 4.7, for all records, it automatically generates the ISO 2709 equivalent of the input data. This data is, then, imported into a predefined *CDS/ISIS for Windows* database.

The appearance of a record, such as that shown in Figure 4.2, in the *CDS/ISIS for Windows* database and formatted by the display format of the software is shown in Figure 4.8.

JournalTitle	: Ethiopian Journal of development Research
Volume	: 20
IssueNumber	:1
Date	:April 1998
ArticleTitle	:THE INFLUENCE OF SELECTED SOCIAL AND DEMOGRAPHIC FACTORS ON FERTILITY : THE CASE OF BAHIRDAR TOWN
Author	: Eshetu Wencheko' and Habtamu Ashenafi"
Abstract	: In this paper an attempt has been made to assess the influence of some socio-demographic variables on ftrtility in Bahirdar, the capital of the Amhara Regional Government Among the variables considered in the study, it was found out that women employment was inversely related to fertility, though this relationship was found to be not significant in the Younger and older age groups. ne educational level of wife and husband also appeared to have an inverse relationship with fertility only in certain age groups. It was also observed that income had a positive effect on fertility. nere was no significant difference in fertility between Moslems and Christians, and between Amharas and other ethnic groups. The findings for Bahirdar were expected to indicate the types of measures needed to be taken in order to challenge the, problem of population growth, especially in townships which may also be used in comparative studies of similar nature.

Figure 4.8. The *Ethiopian Journal of Development Research* in CDS/ISIS for Windows display format

CHAPTER FIVE: CONCLUSION AND RECOMMENDATIONS

5.1. Conclusion

In search for better life, mankind has been doing all his best to utilise the potentials that exist in our universe. This investigation has brought about several research and development communities around the globe.

To keep the continuity of research and development, it was a must to build a certain forum of communication among scientists. The viable mechanism discovered to meet such communication demand was to publicise results of research in journals.

As the number of disciplines and the number of researchers in each field of study increased from time to time, however, the number of research reports also proliferated at a rate that couldn't be controlled easily. In response, it has been realised that the principle of the universal bibliographic control of journal articles is the real solution to the problem. The core principle for the international control of journal articles gives the answer to the question: what has been published in a given subject, in a given area, when and who produced it? Practising the principle of the universal bibliographic control of journals demanded for the establishment of national nodes that keep track of research and development publications in a given nation. These institutions are in charge of collecting, organising and publicising all knowledge output in a given nation.

With the objective of meeting the national bibliographic control of Ethiopian publications, Ethiopia has established a national bibliographic service centre. The centre is named Legal Deposit and National Bibliography Team (LDANBT), which is affiliated to the Ethiopian

National Archives and Library Agency. The findings of this thesis show that there is no any bibliographic publication that can give article level access to the country's journal publications. The absence of such a tool is a drawback both for the nation and the world-wide community. In an age when scientific investigations are increasingly becoming available in machine-readable form and are made accessible to people world-wide within a fraction of a minute, it is unfortunate that the LDANBT can't make such a tool, even the printed version. It is this vital importance of an electronic version of a bibliographic tool for Ethiopian journal articles that motivated the researcher to launch the present research.

To utilise the benefits of OCR technology in bibliographic record production out of printed documents, attempt has been made to develop a system called *Ethiopian Articles Recording System (EARS)* that can capture bibliographic elements from the title page of printed articles in journals published in Ethiopia. The system mainly basis itself on the principles of document analysis and document understanding, which depend on the nature of documents under study. Both principles are built on the geometric and non-geometric features of journal articles published in Ethiopia.

The core component of the system is the document analysis part that is composed of two levels of segmentation. The first level segmentation creates four text zones (various fields belonging to the first text zone, article title, author (s) and author abstract), while the second partitions the first text zone into journal title, volume, issue number, year and page range.

Based on the results of document analysis, the document understanding part of the system considers the geometric features of documents particularly zone order to classify article title, author (s) and author abstract. It also considers non-geometric features such as punctuation

marks in order to classify the fields in the first text zone into journal title, volume, issue number, year and page range.

The system is tested on 97 articles, and it is 85.57% successful. The main sources of error for the system are attributed to the OCR software which created a number of mis-recognised or rejected characters that constitute the keywords for journal identification and several punctuation marks that serve as the non-geometric features for field segmentation and classification. It has also introduced unnecessary white line (that doesn't exist in the original document) in the article title zone for large font article titles.

It is learned that the automatic extraction of bibliographic elements for articles from printed journals published in Ethiopia is technically feasible. However, it is easier to realise that the success of document analysis and document understanding for journal articles published in Ethiopia mainly depend on the accuracy of the OCR software; the OCR software should never mis-recognise at least keywords and punctuation marks. Hence, there is a need to use better and more reliable OCR software to get a better result, and improve the latter process of searching, after the output of *EARS* has been formatted to a text retrieval system such as *CDS/ISIS for Windows*.

5.2. Recommendations

To fully utilise the potentials of automated (OCR-based) data entry for the generation of electronic records out of printed journal article title pages for Ethiopia, a number of things have to be kept in mind and required actions should be made. The main requirements are highlighted below.

1. The LDANBT of the Ethiopian National Archives and Library Agency should materialise one of its main objectives: production of national journal index. To this end, the results of the present research would serve as a starting point.
2. From the discussions held with the Head of Technique and Computer Services, it is learned that they are happy to exercise the method of automatically producing national journal index for Ethiopia. The head noted that his office would collaborate and closely work with the staff in the LDANBT in order to materialise this goal. On the other hand, the Head of LDANBT admits that her office doesn't have personnel that have the expertise of computing. But, her office can do all kinds of support to the staff of Technique and Computer Services in producing the national journal index. The heads of both offices said that the two offices have the culture of working in collaboration. This has been shown in the past, for instance, in the production of the electronic National Bibliography out of the printed version. Hence, any amendments needed for the source code of the present research would be handled with out any need for additional manpower and material resources (at least computers). The computers are so powerful (see table 1.2, chapter 1) that they meet the requirements for handling Microsoft Visual C++ 6.0 (at least 16MB RAM, MS Windows 95 or beyond operating system). There is a programmer to handle any modifications needed for the source code of the system developed in the present research. The only requirements are to have Microsoft Visual C++ 6.0 and latest

and more powerful OCR software that can alleviate the problems observed in *WordScan 3.0* (unable to maintain the original content of the source document). The Head of Technique and Computer Services assures that there is no problem to have MS Visual C++ 6.0 and any OCR software in place. In addition, they need to have an interface program such as *Fangorn* in order to format data into ISO 2709 format so that it can be manipulated by *CDS/ISIS for Windows.*

3. Most of the journals published in Ethiopia don't have the required fields for bibliographic record production in their article title page. This places a challenge of automating national journal index production. Hence, the Team has to inquire journal publishers to meet the requirement of including all fields required for journal indexing in an article title page. This can be enforced when journals are registered for copyright and accredited to deserve ISSN number. The same strategy can be used to force journal publishers in the country to be consistent in the order of presentation of various bibliographic fields in the article title page of journals. This is because many layout variations mean much effort for program development to handle the variation. Therefore, if publishers are required to follow certain standard layouts in presenting bibliographic fields in an article title page, it means that the life of the programmer to automate the production of the bibliographic fields becomes easier.

4. For journals that don't have author abstract in their article title page, another study tailored towards the development of a system to capture article contributions from the table of contents of journal issues should be made.

5. Several studies in the development of Amharic OCR are already underway here at the School of Information Studies for Africa, Addis Ababa University. Therefore, studies should be conducted to capture the bibliographic fields of journal articles in the Amharic language. This study can immediately follow the development of Amharic OCR that can handle real problems.

6. The journal identification database has to be expanded to incorporate the identification keyword for newly emerging journals. In addition, feature functions for these journals have to be incorporated in the program.

7. As already stated in chapter three, time limit doesn't permit the researcher to consider journals the fields in the first zone of which is at the bottom of an article title page for program development. For practical applications, therefore, the program developed in the present study should be modified to handle these groups of journals.

8. The output data of the program, after it is imported into *CDS/ISIS for Windows* database, has to be made accessible to the world-wide community connected to the Internet. Incorporating the database as part of NALA's effort for web-site development can make this.

BIBLIOGRAPHY

Besemer, Hugo and Paul Nieuwenhysen (1992). *How to Use Fangorn – a Microcomputer Programme for the Conversion of Databases to ISO-2709 format.*

CEB, NLM (1999). *Medical Article Record System (MARS), MD.* **http://archive.nlm.nih.gov/proj/mars/mars.html**

Collison, Robert (1969). "Bibliographic Service Center," In: *Encyclopedia of Library and Information Science* edited by Allen Kent & Harold Lancour. Vol.2, New York: Marcel Dekker. Pp.381-384.

Endashaw Bekele (1995). *Current Status of Research Development Problems and Management in Higher Education in Ethiopia with Particular Reference to Addis Ababa University*. Addis Ababa: Research and Publications Office, Addis Ababa University.

Engida Hailye (1999). *Strategic Educational Management Information System for Higher Institutions: The Case of Addis Ababa University*. M.Sc. Thesis (unpublished). Addis Ababa University.

NALE (1998). *Ethiopian Periodicals and Non-Book Publication Index*. Vol.16.

NALE (1999). *Ethiopian Periodicals and Non-Book Publication Index*. Vol.17.

NALE (1993). *Felege Tibebe: 50th Anniversary.*

NALE (1997). *The Value of Information Yesterday and Today: a Brochure.*

FDRE (1999). *Ethiopian National Archives and Library Proclamation 179/1999*. Addis Ababa. Pp. 1130-1138.

Ford, Glenn M., Susan E. Hauser, George R. Thoma (1999). "Automatic Reformatting of OCR text from Biomedical Journal Articles," Presented at Sysmposium for Document Image Understanding Technology at Annapolis, MD. **http://archive.nlm.nih.gov/proj/mars/mars.html**.

Gorman, G.E.(1983). "Current National Bibliographies in Developing Countries of the Commonwealth". ***Libri***, 33(3): 177-189.

Gray, Peter J. (1977). "Optical Scanning, OCR and MICR". In: *Automatic Data Processing Handbook*. New York : McGraw-Hill Book Company. Pp. 2-117 - 2-125.

Haigh, S.(1996). *Optical Character Recognition (OCR) as a Digitization Technology*. Network Notes #37. Information Technology Services, National Library of Canada.

Harrison, A.D., F.A. Roos and R.E. Thomas (1995). "Semi-Automatic Capturing of Bibliographic Information from Journal Contents Pages for Inclusion in Online Library Catalogues: The RIDDLE Project". ***The Electronic Library***, 13(1): 15-20.

Jetform UK ltd, (1998). "Banking on JetForm: Phasing out Forms and Cutting Processing Times from Days to Minutes in the US and in Sweden, Case Study". *Information Management & Technology: The Journal of Cimtech,* 31(6): 266.

Kebede Hunde (1997). *Retrospective Conversion of Printed Card Catalogs to Online Catalog Using Optical Character Recognition Technology: the Case of Addis Ababa University Libraries*. M.Sc. Thesis (unpublished). Addis Ababa University.

Kilgour, F.G.(1969). "Computerisation: The Advent of Humanization in the College Library". *Library Trends*, 18(1): 29-36.

Kuny, T.(1995). *An Introduction to Digitisation Technologies and Issues*. Network Notes # 14. Information Technology Services, National Library of Canada.

Le, Daniel et al (1999). "Automated Labelling of Zones from scanned Documents," Presented at Symposium for Document Image Understanding Technology at Annapolis, MD. **http://archive.nlm.nih.gov/proj/mars/mars.html**.

Liang, J, et al. (1996). "The Prototype of a Complete Document Image Understanding System". *International Association for Pattern Recognition. Workshop on Document Analysis System*, Malvern, PA. Pp. 131-154.

Line, Maurice B. (1983). "National Library & Information Planning". *International Library Review*, 15(3): 227-243.

Oram, Derek O. and Anthony B. Ragozzino (1977). "Input Devices". In: *Automatic Data Processing Handbook*. New York: McGraw-Hill Book Company. Pp. 2-103 - 2-116.

Rao, R.I.K.(1990). *Library Automation*. New Delhi: Wiley Eastern Limited.

RPO, AAU(1994). *Information on Locally Published Scholarly Journals*. Addis Ababa: Research and Publications Office, Addis Ababa University.

Saffady, William (1983*). Introduction to Automation for Librarians*. Chicago : American Library Association.

Saffady, William (1994). *Introduction to Automation for Librarians*. Chicago: American Library Association.

Thoma, George R.(1999). "Automated Data Entry into MEDLINE," Presented at Sysmposium for Document Image Understanding Technology at Annapolis. **http://archive.nlm.nih.gov/proj/mars/mars.html**.

Tsujimoto, S. and H. Asada (1992). "Major Components of a Complete Text Reading System". *Proceedings of the IEEE*, 80(7): 1133-1149.

Weibel, S., M. Oskins and D. Vizine-Goetz(1989). "Automated Title Page Cataloguing: A Feasibility Study". *Information Processing & Management*, 25(2): 187-203.

APPENDICES

APPENDIX I: Profile (five volumes per journal title) of locally published journals in Ethiopia which are received by LDANBT

NO.	Journal title	Publisher	Date of first issue	Freq per year	Last volume received	The 2nd from last volume	The 3rd from last volume	The 4th from last volume	The 5th from last volume	Total articles and issues for each journal title Articles	issues
1.	Ethiopian Pharmaceutical Journal	The Ethiopian Pharmaceutical Association	1975	1	1998, v16 5 articles 3 comms.	1997,V15 4 articles 2 comms.	1996,v14 4 articles 3 comms	1995,v13 5 articles 1 comms	1994,v12 6 articles	33	5(%)
2.	Ethiopian Journal of Development Research	Institute of Development Research, AAU	1974	2	1998,v20,n1 3 articles	1997,v19,n1 3 articles	1996,v.18,n1 4 articles	1995,v17,n1 4 articles	1994,v16,n1 3 articles	32	10(100%)
					1998,v20,n2 3 articles	1997,v19,n2 3 articles	1996,v.18,n2 3 articles	1995,v17,n2 3 articles	1994,v16,n2 3 articles		
3.	Ethiopian Journal of Health Development	Ethiopian Public Health Association	1984	3	1998,v12,n1 10 articles 1comm.	1997,v11,n1 12 articles 1comm	1996,v10,n1 10 articles	1995,v9,n1 6 articles 1review article	1994,v8,n1 8 articles	137	15(100%)
					1998,v12,n2 7 articles 1comm 2review arti.	1997,v11,n2 11 articles 1review arti.	1996,v10,n2 9 articles 1 comm	1995,v9,n2 8 articles 1review article	1994,v8,n2 7 articles		
					1998,v12,n3 8 articles	1997,v11,n3 16 articles 1review arti.	1996,v10,n3 7 articles 1review arti.	1995,v9,n3 4 articles 2 review arti. 1 coom	1994,v8,n3 special issue (book format)		
4.	Ethiopian Medical Journal	Ethiopian Medical Association	1962	4	1999,v37,n1 5 articles 2 comms	1998,v36,n1 5 articles 1 comm 1 case report	1997,v35,n1 5 articles 1 comm 1 case report	?	1995,v33,n1 6 articles 1 comm 1 case report	115	17(85%)
					1999,V37,N2 4 articles 1 comm 2 case reports	?	1997,v35,n2 4 articles 1 comm 2 case reports	1996,v34,n2 4 articles 1 comm 1 case report	1995,v33,n2 5 articles 1 comm 1 case report		
					1999,v37,n3 4 articles 1 comm 1 case report	1998,v36,n3 4 articles 2 case reports	1997,v35,n3 4 articles 1 comm 1 case report	?	1995,v33,n3 4 articles 2 case report		

					1999,v37,n4 7 articles	1998,v36,n4 5 articles 1 comm 1 case report	1997,v35,n4 3 articles 2 comms 3 case reports	1996,v34,n4 5 articles 1 case report	1995,v33,n4 5 articles 1 comm 1 case report		
5.	The Ethiopian Journal of Education	Institute of Education Research, AAU	1967	2	1998,v18,n1 4 articles	1997,v17,n1 3 articles	?	1995,v15,n1 3 articles	1994,v14,n1 3 articles	25	8 (80%)
					1998,v18,n2 4 articles	1997,v17,n2 2 articles	?	1995,v15,n2 3 articles	1994,v14,n2 3 articles		
6.	Journal of Ethiopian Studies (bilingual)	Institute of Ethiopian Studies, AAU	1963	2	?	1998,v31,n1 5 articles	1997,v30,n1 3 articles	?	1995,v28,n1 4 articles 3 review articles	29	8(80%)
					1999,v32,n2 3 articles	1998,v31,n2 2 articles	1997,v30,n2 2 articles	1996,v29,n2 3 articles	1995,v28,n2 4 articles		
7.	Ethiopian Journal of Health Sciences	Jimma Institute of Health Sciences	1990	2	1999,v9,n1 5 articles 3 comm 1 case report	1998,v8,n1 5 articles 4 comm	1997,v7,n1 6 articles 1case report	?	1995,v5,n1 6 articles	61	8(80%)
					1999,v9,n2 6 articles 3 comm	1998,v8,n2 6 articles 2 comm 1 case report	1997,v7,n2 6 articles 1case report	1996,v6,n2 4 articles 1 comm	?		
8.	Ethiopian Journal of Economics	The Ethiopian Economic Association	1992	2	1996,v5,n1 4 articles	1995,v4,n1 4 articles	1994,v3,n1 4 articles	1993,v2,n1 4 articles	1992,v1,n1 4 articles	36	9(90%)
					?	1995,v4,n2 4 articles	1994,v3,n2 4 articles	1993,v2,n2 4 articles	1994,v1,n2 4 articles		
9.	SINET: An Ethiopian Journal of Science	Faculty of Science, AAU	1978	2	?	1998,v21,n1 8 articles	1997,v20,n1 8 articles 2 cooms	1996,v19,n1 6 articles 2 comms	1995,v18,n1 7 articles 2 comms	86	9 (90%)
					1999,v22,n2 7 articles 4 comms	1998,v21,n2 5 articles 7 comms	1997,v20,n2 7 articles 5 comms	1996,v19,n2 7 articles 1 comm	1995,v18,n2 7 articles		

10.	IER Flambeau	Institute of Education Research, AAU	1993	2	1999,v7,n1 6 articles	1998,v6,n1 6 articles	1998,v5,n2 7 articles	?	1995,v3,n2 5 articles	44	8(80%)
					1999,v6,n2 5 articles	1998,v5,n1 5 articles	1997,v4,n1 4 articles	?	1994,v3,n1 6 articles		
11.	Ethiopian Association of Civil Engineers Bulletin	Ethiopian Association of Civil Engineers	1998	2	1998,v1,n1 9 articles	x	x	x	x	9	1(50%)
					?	x	x	x	x		
12.	Journal of Ethiopian Medical Practice	The Ethiopian Society of General Medical Practice	1999	2	1999,v1,n1 2 articles 4 comms 1 case report	x	x	x	x	7	1(50%)
					?	x	x	x	x		
13.	Water : Ethiopian Journal of Water Science and Technology	Arbaminch Water Technology Institute	1997	2	1997,v1,n1 7 articles	x	x	x	x	21	2(100%)
					1997,v1,n2 14 articles	x	x	x	x		
14.	Association of Ethiopian Architects Journal (bilingual)	Association of Ethiopian Architects (AEA)	1997	1	1997,v1 9 articles	x	x	x	x	9	1(100%)
15.	ZEDE: Journal of Ethiopian Association of Engineers & Architects	Ethiopian Association of Engineers and Architects	1965	1	1998,v15 7 articles	1997,v14 6 articles	1996,v13 5 articles 1 technical note	1995,v12 4 articles 1 technical note	1994,v11 5 articles	29	5(100%)
16.	Journal of Ethiopian Veterinary Association	Ethiopian Veterinary Association	1997	2	1998,v2,n1 4 articles	1997,v1,n1 4 articles 2 comms	x	x	x	10	5(100%)
					?	?	x	x	x		

17.	Journal of the Ethiopian Society of Mechanical Engineers	Ethiopian Society of Mechanical Engineers	1997	2	? 1998,v2,n2 6 articles	? 1997,v1,n2 5 articles	x x	x x	x x	11	2(50%)
18.	Bulletin of the Chemical Society of Ethiopia	Chemical Society of Ethiopia	1987	2	1998,v12,n1 9 articles 3 comms 1998,v12,n2 8 articles 1 comm	1997,v11,n1 7 articles 4 comms 1997,v11,n2 3 articles 1 comm	1996,v10,n1 8 articles 3 review articles 1996,v10,n2 6 articles 2 comms	1995,v9,n1 8 articles 1995,v9,n1 5 articles 2 comms	? 1994,v8,n2 6 articles 1 comm	77	9(90%)
19.	Educational Journal (bilingual)	Public Relations Service, Ministry of Education	1995	2	1999,v4,n8 2 articles 1999,v4,n9 2 articles	1998,v4,n7 2 articles ?	1997,v4,n6 3 articles ?	? ?	1996,v2,n3 4 articles ?	13	5(50%)
20.	Eastern Africa Social Science Research Review	Organisation for Social Science Research in Eastern Africa	1985	2	1997,v13,n2 3 articles ?	1996,v12,n1 4 articles 1996,v12,n2 4 articles	1995,v11,n1 5 articles ?	1994,v10,n1 3 articles 1994,v10,n2 4 articles	1993,v9,n1 4 articles 1993,v9,n2 4 articles	31	8(80%)
21.	HISSAB : An Ethiopian Journal of Mathematics (bilingual)	The Mathematical Association of Ethiopia	1964	2	1999,v20,n1 5 articles ?	1997,v19,n1 5 articles 1998,v19,n2 5 articles	1995,v18,n1 (All amharic articles) 1997,v18,n2 (special issue ,book format)	1995,v17,n1 3 articles 1995,v17,n2 3 articles	1993,v16,n1 7 articles 1993,v16,n2 4 articles	32	9(90%)
22.	WALIA: Journal of Ethiopian Wildlife and Natural History Society	Ethiopian Wildlife and Natural History Society	1969	1	1997,n18 7 articles	1996,n17 6 articles	1995,n16 6 articles	1994,n15 6 articles	?	25	4(80%)
23.	Research Bulletin of the Gondar College of Medical Sciences	Gondar College of Medical Sciences	1998	2	1998,v1,n1 8 articles ?	X X	x x	x x	x x	8	1(50%)

APPENDIX II: List of journals based on the number and order of presentation of fields in their article title page

No.	Article title page content and layout/presentation	List of journals
1.	Article title Author(s) (no other attribute)	1.Ethiopian Journal of Languages & Literature 2.Journal of Ethiopian Law
2.	Journal title Article title Author(s) (no abstract)	1. HISSAB : An Ethiopian Journal of Mathematics
3.	Article title Author(s) (no abstract) Journal title volume number (back page)	1. Ethiopian Association of Civil Engineers Bulletin
4.	Article title Author(s) (no abstract) Journal title volume (bottom of page)	1. WALIA : Journal of the Ethiopian Wildlife and Natural History Society
5.	Article title Author(s) (no abstract) Journal Title, number, month year(bottom of page)	1. Association of Ethiopian Architects Journal
6	Journal Title, number, month year Article title Author(s) (no abstract)	1.IER Flambeaw 2.Journal of Ethiopian Studies 3.Research Bulletin of the Gondar College of Medical Sciences
7	Article title Author(s) Abstract (no other attribute)	1. Journal of the Ethiopian Society of Chemical Engineers 2. Educational Journal
8	Article title Author(s) Abstract Journal Title (back page)	1. Ethiopian Journal of Health Development
9	Article title Author(s) Abstract Journaltitle/volume, number (bottom page)	1. Journal of the Ethiopian Veterinary Association
10	Journal title Article title Author(s) Abstract	1. Ethiopian Pharmaceutical Journal
11	Article title Author(s) Abstract Journal title volume(no)/ month year (bottom page)	1. Water : Ethiopian Journal of Water Science and Technology
12	Author, year. Journal title, volume Article title Author(s) Abstract	1. Ethiopian Medical Journal
13	Journal title, volume, number, year(back of page) Article title	1. Ethiopian Journal of Agricultural Economics

	Author(s) Abstract	
14	Journal title, vol. , num., month year Article title Author(s) Abstract	Ethiopian Journal of Development Research The Ethiopian Journal of Education
15	Article title Author(s) Abstract Journal title, volume, number, month year (back of page)	Eastern Africa Social Science Research Review
16	Article title Author(s) Abstract Journal title, vol., num.,year/Journal title (abbreviated) (bottom page)	1. Journal of Ethiopian Medical Practice
17	Article title Author(s) Abstract Journal title, vol., num. , month, year (back top page)	1. Journal of the Ethiopian Society of Mechanical Engineers (ESME Journal) 2. Ethiopian Journal of Health Sciences
18	Journal title, vol.(num.):page range, year Article title Author(s) Abstract	1. SINET : Ethiopian Journal of Science
19	Journal title, volume: page range (year) Article title Author(s) Abstract	1. Ethiopian Journal of Agricultural Sciences
20	Journal title volume, page range Month year Article title Author(s) Abstract (SUMMARY)	1. JESA (Journal of Ethiopian Statistical Association)
21	Journal title, vol., num., month year , page range Article title Author(s) Abstract	1. Ethiopian Journal of Economics
22	Journal title, year ,vol(num), page range Article title Author(s) Abstract	1. Bulletin of the Chemical Society of Ethiopia
23	Article title Author(s) Abstract Journal title, vol.(num.),page range (year) (bottom page)	1.Pest Management Journal of Ethiopia
24.	Article title Author (s) Abstract Text Journal title, volume, year (bottom page)	1. ZEDE: Journal of Ethiopian Architects and Engineers Association

APPENDIX III: Partial Microsoft Visual C++ Source Code for *EARS*

Header file

```
// EARSView.h : interface of the CEARSView class
//
/////////////////////////////////////////////////////////////////////////////

#if
!defined(AFX_EARSVIEW_H__9BFA572C_313D_11D4_929D_00104BC4543C__INCLUDED_)
#define AFX_EARSVIEW_H__9BFA572C_313D_11D4_929D_00104BC4543C__INCLUDED_

#if _MSC_VER > 1000
#pragma once
#endif // _MSC_VER > 1000

struct tree
{
    char word[20];
    tree *left;
    tree *right;
};
struct database
{
    char dbword[50];
    int code;
};

class Node
{
private:
    char journal[150];
    char volume[20];
    char issue[20];
    char year[20];
    char page [20];
    char articletitle[250];
    char author[160];
    char abstract[4000];
public:
    Node();
    ~Node();
        void savedata();
         void setdata(char j[150], char v[20],char i[20], char y[20], char p[20],char ar[250], char
au[160], char ab[4000]);
    char *getjournal();
    char *getvolume();
    char *getissue();
    char *getyear();
    char *getpage();
    char *getarticletitle();
    char *getauthor();
    char *getabstract();
};
class CEARSView : public CEditView
{
```

```
public:
    char keyword[50], *kw;
    char buffer11[50],*jt;//journal title
    char buffer12[20],*vo;//volume
    char buffer13[20],*in;//issue number
    char buffer14[20],*yr;//year
    char buffer15[20],*pg;// page
    char  buffer1[150], *b1; char buffer[150], *b;
    char  buffer2[250]; int buf2linecount;
    char  buffer3[160]; int buf3linecount;
    char  buffer4[4000];int buf4linecount;
    char kword[20];
        char dbword[50];
        char  arttitle[250], *art;
    char artt[1], *ar;
    char  authors[160],*aut;
    char auth[1],*au;
    char  auabstract[4000],*abs;
    char aua[1], *ab;
    int kwcharcount;
    int wordcount;
        int ch,trimcount;
    int charcount,charcount1,charcount2,charcount3, charcount4;
    int b11,b12,whiteline1,b21,b22,whiteline2,b31,b32,whiteline3;
tree *root, *current;
tree* r;// temporary node
database dbase;
Node *start; Node node;
Node *end;
protected: // create from serialization only
    CEARSView();
    DECLARE_DYNCREATE(CEARSView)
// Attributes
public:
    CEARSDoc* GetDocument();
// Operations
public:
    void keywordextracter();
 void createdbase();
 void loadtree();
 int searchinpreorder();
    void Jdevelopment();
    void Jmed();
    void Jagric();
    void Jphar();
    void Jeconomics();
    void Jeducation();
    void Jchem();
    void Jsinet();
    void Jjesa();
// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CEARSView)
    public:
    virtual void OnDraw(CDC* pDC);  // overridden to draw this view
    virtual BOOL PreCreateWindow(CREATESTRUCT& cs);
```

```
    protected:
    virtual BOOL OnPreparePrinting(CPrintInfo* pInfo);
    virtual void OnBeginPrinting(CDC* pDC, CPrintInfo* pInfo);
    virtual void OnEndPrinting(CDC* pDC, CPrintInfo* pInfo);
    //}}AFX_VIRTUAL
// Implementation
public:
    virtual ~CEARSView();
#ifdef _DEBUG
    virtual void AssertValid() const;
    virtual void Dump(CDumpContext& dc) const;
#endif
protected:
// Generated message map functions
protected:
    //{{AFX_MSG(CEARSView)
    afx_msg void OnEarsSegment1();
    afx_msg void OnEarsSegment2();
    afx_msg void OnEarsDisplay();
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
#ifndef _DEBUG  // debug version in EARSView.cpp
inline CEARSDoc* CEARSView::GetDocument()
   { return (CEARSDoc*)m_pDocument; }
#endif
////////////////////////////////////////////////////////////////////////
//{{AFX_INSERT_LOCATION}}
// Microsoft Visual C++ will insert additional declarations immediately before the previous line.
#endif //
!defined(AFX_EARSVIEW_H__9BFA572C_313D_11D4_929D_00104BC4543C__INCLUDED_)
```

Implementation file

```
// EARSView.cpp : implementation of the CEARSView class
//
#include "stdafx.h"
#include "EARS.h"
#include "EARSDoc.h"
#include "EARSView.h"
#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif
////////////////////////////////////////////////////////////////////////
// CEARSView
IMPLEMENT_DYNCREATE(CEARSView, CEditView)
BEGIN_MESSAGE_MAP(CEARSView, CEditView)
    //{{AFX_MSG_MAP(CEARSView)
    ON_COMMAND(ID_EARS_SEGMENT1, OnEarsSegment1)
    ON_COMMAND(ID_EARS_SEGMENT2, OnEarsSegment2)
    ON_COMMAND(ID_EARS_DISPLAY, OnEarsDisplay)
    //}}AFX_MSG_MAP
    // Standard printing commands
    ON_COMMAND(ID_FILE_PRINT, CEditView::OnFilePrint)
```

```
    ON_COMMAND(ID_FILE_PRINT_DIRECT, CEditView::OnFilePrint)
    ON_COMMAND(ID_FILE_PRINT_PREVIEW, CEditView::OnFilePrintPreview)
END_MESSAGE_MAP()
/////////////////////////////////////////////////////////////////////////////
// CEARSView construction/destruction
CEARSView::CEARSView()
{
    // TODO: add construction code here
for (int i = 0; i <=50; i++)
     keyword[i] = 0;
 for (i = 0; i<=50; i++)
  dbword[i] = 0;
 for (i = 0; i<=20; i++)
 kword[i] = 0;



for (i = 0; i<=50; i++) //journal title
    buffer11[i] = 0;
for (i = 0; i<=20; i++)//volume
     buffer12[i] = 0;
for (i = 0; i<=20; i++)//issue number
   buffer13[i] = 0;
for (i = 0; i<=20; i++) //year
    buffer14[i] = 0;
for (i = 0; i<=20; i++)// page
     buffer15[i] = 0;
for ( i = 0; i<=150; i++)
      buffer1[i] = 0;
for (i = 0; i<=150; i++)
     buffer[i] = 0;
for (i = 0; i<=250;i++)
     buffer2[i] = 0;
for (i = 0; i <=160; i++)
      buffer3[i] = 0;
for (i = 0; i <= 4000; i++)
     buffer4[i] = 0;
 ch = trimcount = 0;
 charcount = charcount1= charcount2 = charcount3 = charcount4 = 0;
 b11 = b12 = whiteline1 = 0;
 b21 = b22 = whiteline2 = 0;
 b31 = b32 = whiteline3 = 0;
 wordcount = 0;
 b1 = buffer1;
 jt = buffer11;
vo = buffer12;
in = buffer13;
yr = buffer14;
pg = buffer15;
kwcharcount = 0;
for (i = 0; i<=250; i++)
      arttitle[i] = 0;
art = arttitle;
 for (i = 0; i<=160; i++)
     authors[i] = 0;
 aut = authors;
```

```
    ON_COMMAND(ID_FILE_PRINT_DIRECT, CEditView::OnFilePrint)
    ON_COMMAND(ID_FILE_PRINT_PREVIEW, CEditView::OnFilePrintPreview)
END_MESSAGE_MAP()
/////////////////////////////////////////////////////////////////////////////
// CEARSView construction/destruction
CEARSView::CEARSView()
{
    // TODO: add construction code here
for (int i = 0; i <=50; i++)
     keyword[i] = 0;
 for (i = 0; i<=50; i++)
  dbword[i] = 0;
  for (i = 0; i<=20; i++)
  kword[i] = 0;



for (i = 0; i<=50; i++) //journal title
    buffer11[i] = 0;
for (i = 0; i<=20; i++)//volume
     buffer12[i] = 0;
for (i = 0; i<=20; i++)//issue number
   buffer13[i] = 0;
for (i = 0; i<=20; i++) //year
    buffer14[i] = 0;
for (i = 0; i<=20; i++)// page
     buffer15[i] = 0;
for ( i = 0; i<=150; i++)
      buffer1[i] = 0;
for (i = 0; i<=150; i++)
     buffer[i] = 0;
for (i = 0; i<=250;i++)
     buffer2[i] = 0;
for (i = 0; i <=160; i++)
      buffer3[i] = 0;
for (i = 0; i <= 4000; i++)
     buffer4[i] = 0;
 ch = trimcount = 0;
 charcount = charcount1= charcount2 = charcount3 = charcount4 = 0;
 b11 = b12 = whiteline1 = 0;
 b21 = b22 = whiteline2 = 0;
 b31 = b32 = whiteline3 = 0;
 wordcount = 0;
 b1 = buffer1;
 jt = buffer11;
vo = buffer12;
in = buffer13;
yr = buffer14;
pg = buffer15;
kwcharcount = 0;
for (i = 0; i<=250; i++)
      arttitle[i] = 0;
art = arttitle;
 for (i = 0; i<=160; i++)
     authors[i] = 0;
 aut = authors;
```

```
  for (i = 0; i<=4000; i++)
     auabstract[i] = 0;
  abs = auabstract;
 for (i = 0; i<=1; i++)
     artt[i] = 0;
  ar = artt;
  for (i = 0; i<=1; i++)
      auth[i] = 0;
  au = auth;
  for (i = 0; i <=1; i++)
     aua[i] = 0;
  ab = aua;
  root = current = r = NULL;
}
CEARSView::~CEARSView()
{
}
BOOL CEARSView::PreCreateWindow(CREATESTRUCT& cs)
{
    // TODO: Modify the Window class or styles here by modifying
    //  the CREATESTRUCT cs
    BOOL bPreCreated = CEditView::PreCreateWindow(cs);
    cs.style &= ~(ES_AUTOHSCROLL|WS_HSCROLL);          // Enable word-wrapping
    return bPreCreated;
}
/////////////////////////////////////////////////////////////////////////////
// CEARSView drawing
void CEARSView::OnDraw(CDC* pDC)
{
    CEARSDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: add draw code for native data here
    //RECT ClientRect;
    //GetClientRect(&ClientRect);
   //pDC->DrawText(buffer,-1,&ClientRect,DT_LEFT);
}
/////////////////////////////////////////////////////////////////////////////
// CEARSView printing
BOOL CEARSView::OnPreparePrinting(CPrintInfo* pInfo)
{
    // default CEditView preparation
    return CEditView::OnPreparePrinting(pInfo);
}
void CEARSView::OnBeginPrinting(CDC* pDC, CPrintInfo* pInfo)
{
    // Default CEditView begin printing.
    CEditView::OnBeginPrinting(pDC, pInfo);
}
void CEARSView::OnEndPrinting(CDC* pDC, CPrintInfo* pInfo)
{
    // Default CEditView end printing
    CEditView::OnEndPrinting(pDC, pInfo);
}
/////////////////////////////////////////////////////////////////////////////
// CEARSView diagnostics
#ifdef _DEBUG
```

```
void CEARSView::AssertValid() const
{
    CEditView::AssertValid();
}
void CEARSView::Dump(CDumpContext& dc) const
{
    CEditView::Dump(dc);
}
CEARSDoc* CEARSView::GetDocument() // non-debug version is inline
{
    ASSERT(m_pDocument->IsKindOf(RUNTIME_CLASS(CEARSDoc)));
    return (CEARSDoc*)m_pDocument;
}
#endif //_DEBUG
////////////////////////////////////////////////////////////////////////////
// CEARSView message handlers
void CEARSView::OnEarsSegment1()
{
    // TODO: Add your command handler code here
    FILE * f; int fs = 0;
    CFileDialog dlg(true,"*","*.*");
    if(dlg.DoModal()==IDOK)
    {
            if((f = fopen(dlg.GetPathName(), "r")) == NULL)
                    MessageBox ("File could not be opened");
    }
    else
            return;
//////////// detection of zone one ////////////////////////////////////
ch = fgetc(f);
while ((ch ==32) ||(ch==10))
{
    ch = fgetc(f); trimcount++;
}
//////////////////////////////////
 buffer1[charcount] = (char) ch;
 charcount++;
 do
 {
     b11 = ch;
     b12 = fgetc(f);
     buffer1[charcount] = (char) b12;
     charcount++;
      if((b11!=10) && (b12 ==10))
     {
              ch = fgetc(f);
              if (ch == 10) whiteline1++;
              buffer1[charcount] = (char)ch;
              charcount++;
      } else if ((b11 ==10) && (b12 == 10)) whiteline1++;
    ch = fgetc(f);
    buffer1[charcount] = (char) ch;
     charcount++;
 } while (whiteline1 == 0);
charcount1 = charcount + trimcount; charcount = trimcount = 0;
 strcpy(buffer,buffer1);
```

```
 ////////////////// detection of zone two ////////
 fs = fseek(f, charcount1   ,SEEK_SET);
  if(fs) MessageBox("Fseek one failed");
  //else AfxMessageBox("File pointer is set to first character of field two");
 ////////////// trimmer ////////////////////
ch = fgetc(f);
while ((ch ==32) ||(ch==10))
{
    ch = fgetc(f); trimcount++;
}
//////////////////////////////////
  buffer2[charcount] = (char) ch;
  charcount++;
  do
  {
      b21 = ch;
      b22 = fgetc(f);
      buffer2[charcount] = (char) b22;
      charcount++;
      if((b21!=10) && (b22 ==10))
      {
                ch = fgetc(f);
                if (ch == 10)whiteline2++;
                buffer2[charcount] = (char)ch;
                charcount++;
      } else if ((b21 ==10) && (b22 == 10))whiteline2++;
     ch = fgetc(f);
     buffer2[charcount] = (char) ch;
     charcount++;
  } while (whiteline2 == 0);
 charcount2 = charcount + trimcount; trimcount = charcount = 0;whiteline2 = 0;
//////////////////   detection of field three //////////////
fs = fseek(f, charcount1 + charcount2 + 1, SEEK_SET);
if(fs) MessageBox("Fseek one failed");
//else AfxMessageBox("File pointer is set to first character of field three");
////////////// trimmer ////////////////
ch = fgetc(f);
while ((ch ==32) ||(ch==10))
{
    ch = fgetc(f); trimcount++;
}
buffer3[charcount] = (char) ch;
charcount++;
do
  {
      b31 = ch;
      b32 = fgetc(f);
      buffer3[charcount] = (char) b32;
      charcount++;
      if((b31!=10) && (b32 ==10))
      {
                ch =fgetc(f);
                if(ch==10)whiteline3++;
                buffer3[charcount] = (char) ch;
                charcount++;
      }
```

```
      else if((b31 ==10) && (b32 ==10)) whiteline3++;
    ch = fgetc(f);
     buffer3[charcount] = (char) ch;
     charcount++;
  }while (whiteline3 == 0);
  charcount3 = charcount + trimcount; charcount = trimcount = 0; whiteline3 = 0;
 ////////////////// detection of field four ////////////////////////
fs = fseek(f, charcount1 + charcount2 + charcount3 + 1 , SEEK_SET);
if(fs) MessageBox("Fseek one failed");
//else AfxMessageBox("File pointer is set to first character of field four");
/////// trimmer//////////////////////////////////
ch = fgetc(f);
while ((ch ==32) ||(ch==10))
{
    ch = fgetc(f); trimcount++;
}
while(!feof(f))
  {
     buffer4[charcount] = (char) ch;
     charcount++; ch = fgetc(f);
     if(ch == 10) buf4linecount++;
  }
  charcount4 = charcount;
  MessageBox("First level segmentation completed");
  fclose(f);
}
void CEARSView::OnEarsSegment2()
{
    // TODO: Add your command handler code here
loadtree();
keywordcxtractcr();
}
void CEARSView::createdbase()
{
/*  FILE *db;
    CFileDialog dbdlg(true,"*","*.*");
    if(dbdlg.DoModal()==IDOK){

 if((db = fopen(dbdlg.GetPathName(), "r")) == NULL)
      MessageBox ("db File could not be opened");
 else MessageBox("open database...");
    }
    /*if((db = fopen("B:\\database.dat","r"))== NULL)
              MessageBox("can not open file");*/
    /*strcpy(dbword,"EDUCATION");
          fprintf(db,"%s", dbword);
    strcpy(dbword, "CHEM");
    fprintf(db,"%s",dbword);
    strcpy(dbword, "AGRIC");
    fprintf(db,"%s",dbword);
    strcpy(dbword,"DEVELOPMENT");
    fprintf(db,"%s",dbword);
    strcpy(dbword, "ECONOMICS");
    fprintf(db,"%s",dbword);
    strcpy(dbword, "MED");
    fprintf(db,"%s",dbword);
```

```
    strcpy(dbword, "JESA");
    fprintf(db,"%s",dbword);
    strcpy(dbword, "SINET");
        fprintf(db,"%s",dbword);
    strcpy(dbword, "PHARM");
    fprintf(db,"%s", dbword);
    fclose(db);*/
}
void CEARSView::keywordextracter()
{
tree *returnedword = NULL;
char word[20];
int k = 0;
for (int i = 0; i<=20; i++)
  word[i] = 0;
kw = keyword;
while((*b1!= ' ')&&(*b1!= '.')&&(*b1!= '_')&&(*b1!= '-')&&(*b1!= ':')&&(*b1!=',')&&(*b1!='\n'))
 {
  *kw = *b1;
  kw++;
  b1++;k++;
 }
 int c;
 for (i =0; i<=k; i++)
    {
        c = toupper(keyword[i]);
        word[i] = (char) c;
    }
    strcpy(keyword, word);
 int j =  searchinpreorder();
if(j == 0)
    {
      if(*b1!='\n')
        {
          while((*b1 == ' ')||(*b1== '.')||(*b1== '_')||(*b1== '-')||(*b1== ':')||(*b1==','))
            {
                b1++;
            }
            for (i = 0; i<=50; i++)      keyword[i] = 0;
            keywordextracter();
        }else MessageBox("No keyword found in zone");
    }

else
{

     if(strcmp(current->word,"EDUCATION") == 0)
        {
            MessageBox("Education detected ...");
            Jeducation();
        }
   else if(strcmp(current->word,"CHEM") == 0)
        {
            MessageBox("Chemical society....");
            Jchem();
        }
```

```
else if(strcmp(current->word,"AGRIC") == 0)
        {
                MessageBox("Agriculture...");
     Jagric();
        }
      else if(strcmp(current->word,"DEVELOPMENT") == 0)
        {
      MessageBox("Development...");
           Jdevelopment();
        }
     else if(strcmp(current->word,"ECONOMICS") == 0)
        {
                MessageBox("Economics detected ...");
                Jeconomics();
        }
     else if(strcmp(current->word,"MED") == 0)
        {
                MessageBox("Medical journal detected...");
               Jmed();
        }
     else if(strcmp(current->word,"JESA") == 0)
        {
                MessageBox("JESA detected ...");
                Jjesa();
        }
    else if(strcmp(current->word,"SINET") == 0)
        {
                MessageBox("SINET detected...");
                Jsinet();
        }
  else if(strcmp(current->word, "PHARM") -- 0)
        {
                MessageBox("Phar detected ...");
                Jphar();
        }


////////////// setting and saving data///
/*  char *abstr = buffer4;
    char buffer44[4000],*abstra;
       abstra = buffer44;
    char tempabst[20],*tempa;
    char abstword[20];

    for (int i = 0; i<=20; i++)
        tempabst[i] =0;
    for (i = 0; i<=20; i++)
        abstword[i] =0;
  tempa = tempabst;
  int abscount =0;
   while ((*abstr!=' ')&&(*abstr!='.')&&(*abstr!=':')&&(*abstr!='\n'))
   {
        *tempa = *abstr;
                abscount++;
        tempa++;
        abstr++;
```

```
            }
      int cha;
      for (i =0; i<=abscount; i++)
      {
                  cha = toupper(tempabst[i]);
                  abstword[i] = (char) cha;
      }
 if((strcmp(abstword,"ABSTRACT") == 0)||(strcmp(abstword,"SUMMARY") == 0))
      {
                  while ((*abstr!=' ')&&(*abstr!=':')&&(*abstr!='.')&&(*abstr!='\n'))
                  {
                    *abstr++;
                  }
          while((*abstr == ' ')||(*abstr == ':')||(*abstr == '.')||(*abstr == '\n'))
                  {
                   *abstr++;
                   abscount++;
                  }
      strcpy(buffer44, buffer4);
      } else strcpy(buffer44, buffer4);*/

//////////////////////////////////////////
      node.setdata(buffer11, buffer12, buffer13, buffer14, buffer15, buffer2, buffer3, buffer4);
      node.savedata();
}
current = NULL;
}
void CEARSView::loadtree()
{
   FILE *fdb;
   if((fdb = fopen("G:\\database.dat", "r")) == NULL)
     MessageBox ("Database File could not be opened");
     if((root = new tree) == NULL)
     MessageBox("No value for root");
root->left = NULL;
root->right = NULL;
current = root;
fscanf(fdb,"%s",root->word);
while(!feof(fdb))
{
    if((r = new tree) == NULL)
                MessageBox("no value for r");
    r->left = NULL;
    r->right = NULL;
    fscanf(fdb,"%s",r->word);
                current = root;

while((current->left!=NULL)||(current->right!=NULL))
    {
                if(strcmp(r->word,current->word)<0)
                {
                            if(current->left!=NULL)
                  current = current->left;
                }
                else if(strcmp(r->word,current->word)>0)
                {
```

```
                        if (current->right!=NULL)
                            current = current->right;
                }
    if((current->left==NULL)||(current->right == NULL)) break;
      }
     if(strcmp(r->word,current->word)<0)
     {
                current->left = r;
}
     else
     {
                current->right = r;
     }
}
fclose(fdb);
}
int CEARSView:: searchinpreorder()
{
     bool found = false;
     current = root;
do
{
     if(strcmp(keyword,current->word) == 0) found = true;
     else if(strcmp(keyword,current->word)<0)
                        current = current->left;
     else  current = current->right;
}while((current!=NULL)&&(found == false));
if(found) return 1;
else return 0;
}
void CEARSView::Jdevelopment()
{
  b = buffer;
  while(*b!=',')
  {
       *jt = *b;
       jt++;
       b++;
  }
 while ((*b == ',') ||(*b == ' '))
 {
    *b++;
 }
 ////trimming vol///
 while (*b!=' ')
 {
    *b++;
 }
 while(*b == ' ')
 {
      *b++;
 }
 while((*b!=' ')&&(*b!=','))
 {
      *vo = *b;
      vo++;
```

```
        b++;
  }
  while ((*b == ',') ||(*b == ' '))
  {
     *b++;
  }
  ///// trimming no.
  while ((*b!= '.')&&(*b!= ' '))
  {
     *b++;
  }
 while((*b == '.')||(*b==' '))
 {
     *b++;
 }
 while((*b!=',')&&(*b!='.')&&(*b!=' '))
  {

        *in = *b;
        in++;
        b++;
  }
  while ((*b == ',') ||(*b == ' ')||(*b=='.'))
  {
     *b++;
  }
 while(*b!='\n')
  {
        *yr = *b;
        yr++;
        b++;
  }
}
void CEARSView::Jmed()
{
   int wcount = 0;
   b = buffer;
   while((*b!='.')&&(*b!=' '))
   {
        *yr = *b;
        yr++;
        b++;
   }
while ((*b == '.') ||(*b == ' '))
  {
     *b++;
  }
  while((*b!=',')&&(wcount<3))
  {

        *jt = *b;
        jt++;
        b++;if(*b == ' ') wcount++;
  }
  while ((*b == ',') ||(*b == ' '))
  {
```

```
    *b++;
  }
  while((*b!=',')&&(*b!='.')&&(*b!='\n'))
  {
      *vo = *b;
      vo++;
      b++;
  }
 }
 void CEARSView::Jagric()
 {
      int dotcount = 0;;
      b = buffer;
   while((*b!=',')&&(dotcount<4))
   {
       *jt = *b;
       jt++;
       b++;if (*b == '.') dotcount++;
   }
   while ((*b == ',') ||(*b == ' ')||(*b=='.'))
   {
      *b++;
   }
   while(*b!=':')
   {
       *vo = *b;
       vo++;
       b++;
   }
   while ((*b == ',') ||(*b == ' ')||(*b == ':'))
   {
     *b++;
   }
 while(*b!='(')
   {
       *pg = *b;
       pg++;
       b++;
   }
 while ((*b == ',') ||(*b == ' ')||(*b=='('))
   {
     *b++;
   }
  while((*b!='\n')&&(*b!=')'))
   {
       *yr = *b;
       yr++;
       b++;
   }
 }
 void CEARSView::Jphar()
 {
  b = buffer;
  while(*b!='\n')
   {
       *jt = *b;
```

```
        jt++;
        b++;
    }
}
void CEARSView::Jeconomics()
{
    int wcount = 0;
    b = buffer;
    while((*b!=',')&&(wcount<4))
    {
        *jt = *b;
        jt++;
        b++; if(*b ==' ') wcount++;
    }
while ((*b == ',')||(*b == ' '))
  {
     *b++;
  }
while((*b!=',')&&(*b!='('))
  {
      *vo = *b;
      vo++;
      b++;
  }
while ((*b == ',') ||(*b == ' ')||(*b == '(')||(*b == ')'))
  {
     *b++;
  }
while((*b!=',')&&(*b!=')'))
  {
      *in = *b;
      in++;
      b++;
  }
  while ((*b == ',') ||(*b == ' ')||(*b == ')'))
  {
     *b++;
  }
 while(*b!=',')
 {
      *yr = *b;
      yr++;
      b++;
 }
 while ((*b == ',') ||(*b == ' '))
 {
    *b++;
 }
 while(*b!='\n')
 {
      *pg = *b;
      pg++;
      b++;
 }
}
```

```
void CEARSView::Jeducation()
{
 int wcount = 0;
  b = buffer;
  while((wcount<5)&&(*b!=','))
  {
       *jt = *b;
       jt++;
       b++; if(*b ==' ') wcount++;
  }

 while ((*b == ' ')||(*b == ','))
 {

    *b++;
 }
////// trimming vol.////
 while(*b!='.')
 {
      *b++;
 }
while((*b=='.')||(*b==' '))
 {
      *b++;
 }
/////.....
 while((*b!=',')&&(*b!=' '))
 {
      *vo = *b;
      vo++;
      b++;
 }
while ((*b == ',')||(*b == ' '))
 {
    *b++;
 }
////// trimming No.
while(*b!='.')
 {
      *b++;
 }
while((*b=='.')||(*b==' '))
 {
      *b++;
 }
///////.....
 while((*b!=' ')&&(*b!=','))
 {
      *in = *b;
      in++;
      b++;
 }
 while ((*b == ',') ||(*b == ' '))
 {
    *b++;
 }
```

```
//////.....
 while(*b!='\n')
  {
      *yr = *b;
      yr++;
      b++;
  }
}
///////////////Chem segementation//////////
void CEARSView::Jchem()
{
int dotcount = 0;;
     b = buffer;
   while(dotcount<4)
   {
       *jt = *b;
       jt++;
       b++;if (*b == '.') dotcount++;
   }
  while ((*b == ',') ||(*b == ' ')||(*b=='.'))
  {
     *b++;
  }
while((*b!=',')&&(*b!=' '))
  {
      *yr = *b;
      yr++;
      b++;
  }
while ((*b == ',') ||(*b == ' '))
  {
     *b++;
  }
  while(*b!='(')
  {
      *vo = *b;
      vo++;
      b++;
  }
  while ((*b == ',') ||(*b == ' ')||(*b == '('))
  {
    *b++;
  }
while(*b!=')')
  {
      *in = *b;
      in++;
      b++;
  }
while ((*b == ',') ||(*b == ' ')||(*b == ')'))
  {
    *b++;
  }
while((*b!='.')&&(*b!='\n'))
  {
      *pg = *b;
```

```
//////.....
 while(*b!='\n')
  {
       *yr = *b;
       yr++;
       b++;
  }
}
///////////////Chem segementation//////////
void CEARSView::Jchem()
{
int dotcount = 0;;
      b = buffer;
   while(dotcount<4)
   {
        *jt = *b;
        jt++;
        b++;if (*b == '.') dotcount++;
   }
  while ((*b == ',') ||(*b == ' ')||(*b=='.'))
  {
     *b++;
  }
 while((*b!=',')&&(*b!=' '))
  {
       *yr = *b;
       yr++;
       b++;
  }
 while ((*b == ',') ||(*b == ' '))
  {
     *b++;
  }
  while(*b!='(')
  {
       *vo = *b;
       vo++;
       b++;
  }
  while ((*b == ',') ||(*b == ' ')||(*b == '('))
  {
    *b++;
  }
 while(*b!=')')
  {
       *in = *b;
       in++;
       b++;
  }
 while ((*b == ',') ||(*b == ' ')||(*b == ')'))
  {
    *b++;
  }
 while((*b!='.')&&(*b!='\n'))
  {
       *pg = *b;
```

```
        pg++;
        b++;
  }
}
void CEARSView::Jsinet()
{
   b = buffer;
   while(*b!=',')
   {
        *jt = *b;
        jt++;
        b++;
   }
  while ((*b == ',') ||(*b == ' '))
  {
     *b++;
  }
while(*b!='(')
  {
        *vo = *b;
        vo++;
        b++;
  }
  while ((*b == ',') ||(*b == ' ')||(*b == '('))
  {
    *b++;
  }
  while(*b!=')')
  {
        *in = *b;
        in++;
        b++;
  }
  while ((*b == ',') ||(*b == ' ')||(*b == ')')||(*b == ':'))
  {
    *b++;
  }
while(*b!=',')
  {
        *pg = *b;
        pg++;
        b++;
  }
while ((*b == ',') ||(*b == ' '))
  {
    *b++;
  }
 while(*b!='\n')
  {
        *yr = *b;
        yr++;
        b++;
  }
}
////////////// JESA segmentation ////////////////////
void CEARSView::Jjesa()
```

```
{
   b = buffer;
  while(*b!=' ')
  {
       *jt = *b;
       jt++;
       b++;
  }

 while ((*b == ',') ||(*b == ' '))
 {
    *b++;
 }
while(*b!=',')
 {
      *vo = *b;
      vo++;
      b++;
 }
 while ((*b == ',') ||(*b == ' '))
 {
    *b++;
 }
while(*b!='\n')
 {
      *pg = *b;
      pg++;
      b++;
 }
while ((*b == ',') ||(*b == ' ')||(*b =='\n'))
 {
    *b++;
 }
while(*b!='\n')
 {
      *yr = *b;
      yr++;
      b++;
 }
}
//////// class node message handlers /////////
Node::Node()
{
    strcpy(journal, " ");
    strcpy(volume, " ");
    strcpy(issue, " ");
    strcpy(year, " ");
    strcpy(page, " ");
    strcpy(articletitle, " ");
    strcpy(author, " ");
    strcpy(abstract, " ");
}
Node::~Node()
{

}
```

DECLARATION

This thesis is my original work and has not been submitted for a degree in any other university.

Enchalewyi

ENCHALEW YIFRU AYALEW
19 May 2000

The thesis has been submitted for examination with our approval as university advisors.

Getachew Birru
19 May 2000

Worku Alemu
19 May 2000