MLIS S1-01

ONE YEAR MASTER'S DEGREE COURSE

(CBCS in Accordance with NEP 2020)

SEMESTER - I

INFORMATION STORAGE AND RETRIEVAL

LIBRARY AND INFORMATION SCIENCES

Guru Prasad Das School of Interdisciplinary and Vocational Studies

KRISHNA KANTAHANDIQUI STATE OPEN UNIVERSITY

Subject Experts

Prof Sanjay Kumar Singh, DLIS, Gauhati University

Dr. Dipen Deka, DLIS, Gauhati University

Dr. Deepjyoti Kalita, Cotton University

Course Co-ordinator: Komal Kumari, KKHSOU

SLM Preparation Team

UNITS CONTRIBUTORS

1-13	Ms. Komal Kumari, KKHSOU
14	Mr. Manash Kumar, Swahid Jadav Nath Homoeopathic Medical College & Hospital

Editors

Content & Language:	Dr. Jacqueline J. Thabah, NEHU; Mr. Pranjit Kalita, KKHSOU; Mr. Dibyajyoti Patgiri, KKHSOU

Structure & Format: Ms. Komal Kumari, KKHSOU

August, 2026 *ISBN: 978-93-7839-012-8*

Copyright Krishna Kanta Handiqui State Open University

This Self Learning Material (SLM) of the Krishna Kanta Handiqui State University is made available under a Creative Commons Attribution-Non Commercial-ShareAlike4.0 License (International): http.//creativecommons.org/licenses/by-nc-sa/4.0.

Printed and published by Registrar on behalf of the Krishna Kanta Handiqui State Open University.

Headquarters: Patgaon, Rani Gate, Guwahati-781017City Office: Resham Nagar, Khanapara, Guwahati-781022 Web: www.kkhsou.in

M. Lib. I. Sc

Information Storage and Retrieval

DETAILED SYLLABUS

Unit 1: Introduction to Information and IR **7-18**

Information Retrieval: definition, features, scope and objectives; Need for information retrieval; Brief history of information retrieval and early use of computers in IR.

Unit 2: Basic Concepts and Components of IR Systems **19-29**

Types of IR systems; Basic components IR system; Purpose and functions of IR systems.

Unit 3: Database Management Systems **30-42**

Introduction to Database Management Systems (DBMS); Advantages of DBMS in ISAR context; ACID properties and their relevance to IR databases; Data models: object-based logical model; record-based logical model; DBMS languages: DDL, DML, DCL.

Unit 4: Database modelling system **43-52**

Database modelling system: Linear sequential model; Hierarchical model; Network model; Entity-relationship (E-R) model; Relational model; Object-oriented model.

Unit 5: Organisation of Data **53-63**

Data organisation; Record structure; Organisational methods: Sequential file structure (advantages and disadvantages); Direct file organisation; Index file structure and trees (e.g., B-trees) for fast retrieval.

Unit 6: Indexing in IR **64-77**

Document representations; Query formulation; Query matching, Relational statements in query specification; Relevance feedback

Unit 7: Evaluation and Measurement of IR Systems **78-91**

Evaluation of IR systems; Need for evaluation; Evaluation criteria and approaches; Evaluation of outcome: recall and precision, fallout and generality; limitations of recall and precision; Evaluation experiments: Cranfield tests, MEDLARS, SMART experiments.

Unit 8: Multimedia Information Retrieval **92-103**

Introduction to multimedia information retrieval; concepts and challenges, example; Text information retrieval vs audio and music information retrieval; Image information retrieval; Video information retrieval.

Unit 9: Users of IR Systems and Information Needs **104-117**

Users and their nature; Types of information needs; Information needs in different areas of activity: Industrial; Planning; Decision-making; Research and development; Business; Information seeking behaviour of users; User studies and their role in IR system design and improvement.

Unit 10: Standards and Protocol in IR **118-133**

Information retrieval standards and protocols: Z39.50 (including how Z39.50 works); SRW, SRU, CQL; Intelligent information retrieval: Expert systems (ES) and their applications for library professionals; ES in cataloguing, classification, document delivery, abstracting; Vocabulary control in IR system, Thesaurus.

Unit 11: Statistical Methods for IR **134-147**

Statistical methods in IR; Boolean model; Vector Space Model (VSM); Probabilistic model.

Unit 12: Natural Language Processing **148-160**

Natural language processing in information retrieval; NLP techniques and tasks: Tokenization; Stemming and stemming algorithms; Lemmatization and comparison with stemming, NLP dictionaries; Semantic analysis: knowledge base and knowledge representation using semantic networks and frames.

Unit 13: Web-based Information Retrieval **161-173**

World Wide Web: document model and naming; Users' interaction and search behaviour on the web; Search engines: types, features and advanced search applications; Web directories; Semantic web and its role in IR.

Unit 14: Semantic Web **174-188**

Semantic web components: XML: salient features and issues; RDF and RDF triplets; salient features of RDF; RDF Schema (RDFS): differences from RDF; issues; Ontology; Logic and ontology languages; description logics and components; Web Ontology Language (OWL Full, OWL DL, OWL Lite);

COURSE INTRODUCTION

Welcome to the course on "Information Storage and Retrieval." This course is designed to provide you with a comprehensive understanding of the principles, systems, and technologies that govern how information is organised, stored, and made accessible to users. In an era of exponential information growth, the ability to efficiently store and retrieve relevant information from large collections has become one of the most critical functions of library and information systems. This course will equip you with the theoretical foundations and practical insights necessary to understand, design, evaluate, and improve information retrieval systems across a wide range of environments, from traditional library databases to the modern semantic web.

We begin by exploring the fundamental concepts of information and information retrieval, tracing the historical evolution of the discipline from its early manual origins to the introduction of computers in retrieval operations. You will learn how the term "information retrieval" was first articulated by Calvin Mooers in 1950, and how subsequent decades saw the emergence of pioneering systems and experiments that laid the groundwork for contemporary IR. From there, the course examines the basic concepts and components of IR systems their types, architectural elements, and the purpose and functions they serve in connecting users with the information they seek.

The course then takes you into the domain of Database Management Systems (DBMS), where you will understand how structured storage of information is achieved, maintained, and queried. We will study the ACID properties, data models, and the key DBMS languages DDL, DML, and DCL form the technical backbone of information storage. Building on this, the course introduces various database modelling paradigms, including the hierarchical, network, relational, entity-relationship, and object-oriented models, so that you develop a well-rounded understanding of how information entities and their relationships can be formally represented. The organisation of data at the physical level including record structures, sequential files, direct file organisation, and index structures such as B-trees is also covered, giving you insight into how retrieval efficiency is achieved at scale.

A major focus of the course is on indexing in information retrieval, where you will explore how documents are represented, how queries are formulated, and how matching between queries and indexed documents is performed. You will also study relevance feedback as a powerful mechanism for refining retrieval outcomes. Equally central is the evaluation of IR systems you will engage critically with established metrics such as recall and precision, and examine the landmark evaluation experiments that shaped the field, including the Cranfield tests, MEDLARS, and the SMART experiments.

We will further address the human dimension of information retrieval through a dedicated study of users, their information needs, and their information-seeking behaviour across industrial, research, business, and decision-making contexts. You will understand how user studies contribute directly to the design and improvement of IR systems. On the technical and standards side, you will be introduced to key interoperability protocols such as Z39.50, SRW, SRU, and CQL, as well as the role of expert systems and thesaurus-based vocabulary control in intelligent information retrieval. The statistical models underpinning retrieval the Boolean model, the Vector Space Model, and the Probabilistic model will be examined as the theoretical core of modern search.

We will also explore Natural Language Processing techniques including tokenisation, stemming, lemmatisation, and semantic analysis as the linguistic bridge that allows machines to process and interpret human-language queries with greater accuracy.

By the end of this course, you will have developed a thorough and integrated understanding of the full spectrum of information storage and retrieval from data organisation and retrieval models to evaluation, user behaviour, standards, NLP, and semantic technologies. This course aims to prepare you for a career in library and information science by equipping you with the intellectual tools to think critically about information systems and the professional competence to contribute meaningfully to their design, management, and continued evolution.

UNIT 1: INTRODUCTION TO INFORMATION AND IR

UNIT STRUCTURE

1.1 Learning Objectives
1.2 Introduction
1.3 Information Retrieval: Definition, Features, Scope, and Objectives
    1.3.1 Definition of Information Retrieval
    1.3.2 Features of Information Retrieval
    1.3.3 Scope of Information Retrieval
    1.3.4 Objectives of Information Retrieval Systems
1.4 Need for Information Retrieval
1.5 Brief History of Information Retrieval
    1.5.1 Early Cataloguing and Mechanical Devices
    1.5.2 The Role of Vannevar Bush and Memex
    1.5.3 Early Use of Computers in IR
    1.5.4 Key Milestones
1.6 Let Us Sum Up
1.7 Answer to Check Your Progress
1.8 Model Questions
1.9 References and Further Readings

1.1 LEARNING OBJECTIVES

After going through this unit, the you will be able to:

- Define Information Retrieval (IR) and identify its core features, scope, and objectives.
- Explain the fundamental reasons for the development and growing importance of IR.
- Trace the historical development of IR from early cataloguing practices to the early use of computers.
- Distinguish between IR systems and database management systems.
- Understand key milestones and contributors in the history of IR.

1.2 INTRODUCTION

With the rapid advancement of science and technology in the modern era, the generation of data and information across every discipline has grown at a remarkable pace over the last few decades. The sheer volume, variety, and velocity of information being produced make its effective storage, organization, and retrieval a matter of considerable intellectual and practical concern. Simply generating information is not sufficient; it must be made accessible to users in a timely and relevant manner.

Prior to the era of computerization, records such as documents, manuscripts, and artefacts were maintained in libraries and personal collections. Retrieval was achieved through cataloguing schemes and other localized practices. As the volume of information expanded beyond the capacity of traditional systems, and as computational tools became available, a new field emerged to address these challenges systematically. That field is Information Retrieval (IR).

In this unit we will study foundational understanding of Information Retrieval by examining its definition, features, scope, and objectives, followed by an analysis of why IR is necessary in the contemporary information environment.

1.3 INFORMATION RETRIEVAL: DEFINITION, FEATURES, SCOPE, AND OBJECTIVES

1.3.1 Definition of Information Retrieval

The term "Information Retrieval" was first formally coined by Calvin Mooers at a conference in 1950. Since then, the field has attracted numerous definitions reflecting both its theoretical and applied dimensions.

According to Kowalski and Maybury, "An Information Retrieval System is a system that is capable of storage, retrieval, and maintenance of information." In this context, information may be composed of text (including numeric and date data), images, audio, video, and other multimedia objects.

Information Retrieval includes representation, storage, organization, accessing information which actually meets up the user need. This formulation underscores the user-centric character of IR, its ultimate measure of success is the degree to which a user's information need is satisfied. The field began primarily with unstructured textual data, which, unlike structured databases, lacks a clearly defined schema or set of relational constraints.

1.3.2 Features of Information Retrieval

IR systems exhibit several distinguishing features that set them apart from other information management systems:

- **Natural Language Processing:** IR systems are designed to accept queries in natural language, map them against an indexed term store, and locate relevant documents from a term-document matrix.
- **Relevance Ranking:** After executing a query, the system ranks results using algorithms such as PageRank, so that the most relevant documents appear at the top.
- **Incremental Indexing:** Most IR systems index documents on an incremental basis, meaning web-based crawlers periodically scan the web and update the index to reflect newly added or modified content.

- **Multimedia Capability:** Modern IR systems are not restricted to text; they also support retrieval of images, audio, and video, though textual indexing remains the most functionally complete modality.
- **Precision and Recall:** The two primary performance measures of an IR system are precision (the proportion of retrieved documents that are relevant) and recall (the proportion of all relevant documents that are successfully retrieved).
- **Search Overhead Minimization:** A key functional criterion of any IR system is the minimization of overhead, defined as the time a user spends in all activities leading up to reading a relevant item, including query formulation, query execution, and scanning non-relevant results.

1.3.3 Scope of Information Retrieval

The scope of Information Retrieval has expanded considerably since its early focus on bibliographic text. Based on current research and practice, the scope of IR encompasses:

- Document clustering and categorization
- Classification of documents
- System architecture design
- Information and data visualization
- Relevance ranking of documents
- Semantic linking and filtering
- Web search, personalized IR, enterprise/institutional service-based IR, and domain-specific IR
- Multimedia IR, including image, audio, and video retrieval
- Distributed IR systems and digital libraries
- By nature, IR can also be categorized as web-based IR systems, digital libraries, multimedia IR systems, and distributed IR systems. The development of search engines, such as Google (founded in 1997), represents the most commercially visible application of IR principles.

1.3.4 Objectives of Information Retrieval Systems

According to Kowalski and Maybury, "The general objective of an Information Retrieval System is to minimize the overhead of a user locating needed information." This overhead encompasses query generation, query execution, scanning retrieved results, and reading non-relevant items.
The objectives can be elaborated as follows:

1. **Support of Query Generation:** IR systems must help users overcome the natural obstacles in specifying what they need, including vocabulary mismatch between the user's terms and the document authors' terms, ambiguities in natural language, and limited domain knowledge.
2. **Effective Retrieval:** The system must retrieve items that are relevant to the user's stated need, and rank them in order of likely relevance to reduce the burden of manual filtering.
3. **Presentation of Results:** Retrieval results should be presented in a format that facilitates the user's decision-making, including ranking, clustering, summarization, and link analysis.
4. **Support for Comprehensive or Selective Retrieval:** Depending on context (e.g., a financial analyst versus a student), the system must balance comprehensiveness against the risk of information overload.

Check Your Progress 1

Q1. Who coined the term "Information Retrieval"?

Q2. Match the following:

Column A	Column B
(a) Precision	(i) Proportion of relevant documents retrieved
(b) Recall	(ii) Proportion of retrieved documents that are relevant
(c) DBMS	(iii) Works with structured data using relational algebra

Q3. Fill in the blank:

a. An IR system accepts queries in _____ language.

Q4. Name two types of IR systems categorized by nature.

Q5. True or False

a. IR systems and database management systems are functionally identical

1.4 NEED FOR INFORMATION RETRIEVAL

The development of Information Retrieval as a distinct field was not arbitrary, it arose from a convergence of practical pressures and intellectual imperatives. The need for IR can be understood through the following factors:

i. Explosion in Volume of Documents

The size and number of documents produced across disciplines increased to a level where no traditional cataloguing system could provide adequate technical support. Libraries and archival institutions found their conventional tools insufficient in the face of this growth.

ii. Emergence of New Data Types

Different disciplines, such as Earth Observation, Biotechnology, and Genetics, began producing diverse types of data in multiple file formats with the support of computers. Much of this data is either semi-structured (video, audio) or unstructured (web pages, electronic resources), requiring new methods of indexing, storage, organization, and retrieval that conventional library systems were not designed to handle.

iii. Limitations of Traditional Library Systems

Libraries had limited scope in terms of processing documents, handling diverse electronic resources, and sharing heterogeneous data and information over the internet. The functions of a library, while foundational, were not scalable to the demands of a digitally interconnected information environment.

iv. Growth of the World Wide Web

With the development of the web, organizations began publishing and sharing information in large quantities. This information required preprocessing, filtering, and modelling before it could be made accessible and retrievable. Documents needed to be indexed, scanned, and associated with their bibliographic elements. This created a fundamental technical paradigm shift and necessitated the development of new theories and concepts to handle electronic resources effectively.

v. Inadequacy of Pre-coordinate Indexing

The librarians' traditional approach to indexing was based on pre-coordinate systems, and both the success and efficiency of such indexing depended heavily on classification schemes such as the Colon Classification or the Chain Indexing system developed by Dr. S.R. Ranganathan. However, maintaining a pre-coordinate indexing system requires enormous human labour, lacks a computational or statistical basis, and places the entire retrieval burden on the user's knowledge of the correct index term. Post-coordinate approaches, which are implemented in modern search engines, became necessary to address these limitations.

vi. Business and Communication Imperatives

The web emerged as a platform for universal access to human knowledge and as a channel of communication and information sharing. The importance of IR is particularly felt in the necessity to locate shared information without restrictions, and the field has opened up considerable business opportunities through the web environment.

Check Your Progress 2

Q1. Name the pre-coordinate indexing system developed by Dr. S.R. Ranganathan.

Q2. Fill in the blank:

a. The web emerged as a platform of_____of human knowledge capital.

Q3. Match the following:

Column A	Column B
(a) Pre-coordinate indexing	(i) Used in modern search engines
(b) Post-coordinate indexing	(ii) Based on classification schemes like Colon Classification
(c) Semi-structured data	(iii) Examples include video and audio

Q4. True or False:

a. Traditional library cataloguing systems are sufficient to handle all modern digital data types

1.5 BRIEF HISTORY OF INFORMATION RETRIEVAL

The historical development of Information Retrieval traces a path from ancient cataloguing practices through mechanical devices to the first systematic use of computers for information retrieval tasks.

1.5.1 Early Cataloguing and Mechanical Devices

The approach to managing and organizing large collections of information originated in librarianship. It is generally accepted that cataloguing constitutes the foundational precursor to Information Retrieval as a field.

According to Eliot and Rose, in the 3rd century B.C., the Greek poet Callimachus first created his own cataloguing scheme for managing personal collections. The library at Alexandria (280 B.C.) is reported to have held more than 700,000 documents, though whether any organized retrieval mechanism existed for this collection remains unknown. Similarly, Nalanda University maintained a large library for document storage, but the nature of its retrieval practices is not documented.

The development of mechanical retrieval aids began in the nineteenth and early twentieth centuries. In 1891, Rudolph filed a patent in the US Patent Office for a machine composed of catalogue cards joined together, which could be wound past a viewing window to allow rapid manual scanning of the catalogues. In 1918, Soper filed another patent for a device in which catalogue cards with holes, corresponding to subject categories, were aligned in front of each other; if light was visible through the arrangement, a match was found among entries sharing a particular combination of categories.

Emanuel Goldberg was the first person to work toward a solution for mechanized catalogue searching in the 1920s and 1930s. He developed an optical device that could search for a pattern of dots or letters within catalogues stored on a roll of

microfilm. In 1928, Goldberg patented his invention in the US Patent Office: catalogue entries were stored on a roll of film, a query also on film showed a negative image of the entry being sought, and a light source shone through both rolls onto a photocell. When an exact match was found, all light was blocked to the cell and an image of the match was projected onto a screen or photographic plate.

In 1935, Davis and Draeger conducted further experiments along similar lines using microfilm-based searching. Their work is reported to have influenced Vannevar Bush.

1.5.2 The Role of Vannevar Bush and Memex

Vannevar Bush's influential contribution came with the conceptual development of the "Memex" system in 1945. This vision of a mechanized, memory-extending personal device that could store and retrieve an individual's books, records, and communications using associative trails laid important intellectual groundwork for later developments in hypertext and IR systems. Bush's ideas, though not fully realized in hardware during his lifetime, inspired subsequent researchers considerably.

Following Bush's vision, Radolph Shaw implemented the "Rapid Selector" at the US Department of Agriculture (USDA) library. This machine was developed by engineers at MIT and delivered to the USDA in 1949. It was designed to search through a 2,000-foot reel of film; 72,000 frames were stored on the film, indexed by 430,000 entries. Shaw reported that the Rapid Selector could search at a rate of 78,000 entries per minute.

In 1950, Luhn developed another selector using punch cards, light, and photo-cells, capable of searching over 600 cards per minute. An important feature of this system was its ability to search for patterns of consecutive characters within a longer string.

1.5.3 Early Use of Computers in IR

The first known instance of a computer being used for information retrieval purposes is attributed to 1948, when Holmstrom demonstrated that the Universal Automatic Computer (UNIVAC) was capable of searching for text references attached to subject codes stored on magnetic tapes, processing at 120 words per minute.

During the 1950s, numerous projects related to IR were undertaken in organizations such as General Electric. This period saw the emergence of systematic research into automated indexing and retrieval.

The first Information Retrieval systems, as noted by Kowalski, "originated with the need to organize information in central repositories, e.g., libraries." As computers became commercially available, they were recognized as natural platforms for the storage and retrieval of text. Early Database Management Systems provided an initial framework for electronic manipulation of indexes to information, and libraries migrated their hardcopy catalogue formats into structured databases.

1.5.4 Key Milestones (1950s Onwards)

The latter half of the twentieth century saw a series of pivotal developments that shaped modern IR:

- **1951:** Mortimer Taube introduced Co-ordinate and Uniterm Indexing, which offered a post-coordinate approach to indexing.
- **1957 (Cranfield I):** The Cranfield I study, conducted by ASLIB under C.W. Cleverdon, evaluated several indexing systems including Universal Decimal Classification, Alphabetical Subject Catalogues, and Faceted Classification Schemes, investigating the costs and efficiency of indexing and searching.
- **1962:** Allan Kent published "Information Analysis and Retrieval."
- **Early 1960s:** Gerard Salton proposed the vector space model for Information Retrieval, in which the performance of retrieval and ranking was measured by the cosine coefficient of document and query vectors.
- **1963:** The Weinberg Report ("Science, Government and Information") identified the problem of information transfer and the growing crisis in managing and retrieving large volumes of information. It made separate recommendations for the Technical Community and Government Agencies regarding advanced retrieval techniques.
- **1963-1966 (Cranfield II):** Executed by C.W. Cleverdon, Mills, and Keen, this study evaluated retrieval performance on the basis of recall and precision measures.
- **1960s:** The SMART (System for the Mechanical Analysis and Retrieval of Text) system was developed by Gerard Salton and his team at Cornell University. It incorporated the vector space model, relevance feedback, and Rocchio Classification, and became a landmark system in IR research.
- **1968:** Salton published "Automatic Information Organization and Retrieval." J.W. Sammon also introduced the concept of a visualisation interface integrated into an IR system.
- **1966-1971:** F.W. Lancaster evaluated the MEDLARS (Medical Literature Analysis and Retrieval System), which subsequently gave birth to AIM-TWX and eventually to MEDLINE/PubMed.
- **1975:** Salton published three seminal works: "A Theory of Indexing," "A Theory of Term Importance in Automatic Text Analysis," and "A Vector Space Model for Automatic Indexing," which gave significant impetus to IR research.
- **1978:** The ACM SIGIR Conference was established, which emerged as the apex conference in the IR field.
- **1982:** Belkin, Oddy, and Brooks introduced the concept of ASK (Anomalous State of Knowledge) as a model for understanding user information needs in IR.

- **1982-1988:** The OKAPI model was formulated at the Polytechnic of Central London. "Okapi is a set-oriented ranked output design for probabilistic type retrieval of textual material using inverted index."
- **1989:** Tim Berners-Lee proposed the World Wide Web at CERN Laboratory, an event that fundamentally transformed the scale and context of IR.
- **1992:** The TREC (Text REtrieval Conference) began as part of the TIPSTER text programme, sponsored by the US Defense Department and the National Institute of Standards and Technology (NIST). TREC became a major platform for benchmarking IR system performance.
- **1996:** The PageRank algorithm was developed at Stanford University by Larry Page and Sergey Brin, forming the basis of the Google search engine.
- **1997:** Google Inc. was founded and became the dominant force in the web search domain.
- **2003:** Latent Dirichlet Allocation (LDA), a generative topic model in Natural Language Processing, was developed by David Blei, Andrew Ng, and Michael Jordan. LDA is related to Latent Semantic Indexing (LSI) and probabilistic Latent Semantic Analysis (pLSA), which gained significant use in Search Engine Optimization (SEO).

Check Your Progress 3

Q1. In which year was the ACM SIGIR Conference first established?

Q2. Match the following:

Column A	**Column B**
(a) Vannevar Bush	(i) Vector Space Model
(b) Gerard Salton	(ii) Memex System
(c) Tim Berners-Lee	(iii) World Wide Web
(d) Larry Page and Sergey Brin	(iv) PageRank algorithm

Q3. Fill in the blank:

a. The OKAPI model was developed at the _____of Central London.

Q4. What does ASK stand for in the context of Information Retrieval?

Q5. True or False:

a. Calvin Mooers coined the term "Information Retrieval" in 1968.

1.6 LET US SUM UP

We have studied foundational introduction to Information Retrieval as a field of study and practice. Information Retrieval is concerned with the representation, storage, organization, and retrieval of information in response to user needs, operating primarily on unstructured or semi-structured data. Its core features include natural language querying, relevance ranking, incremental indexing, and multimedia capability. The key performance measures of any IR system are precision and recall. The field's scope has broadened significantly to encompass document clustering, semantic linking, web search, and digital libraries.

The need for IR arose from practical limitations of traditional cataloguing systems in the face of rapid information growth, the emergence of new data types, the expansion of the web, and the inadequacy of pre-coordinate indexing for large-scale digital environments. Historically, IR evolved from ancient cataloguing practices through mechanical optical devices in the early twentieth century, to the first use of computers for text retrieval in 1948. The development of the SMART system, the vector space model, the TREC conference, and the PageRank algorithm represent some of the defining milestones that established IR as a mature scientific field.

1.7 ANSWER TO CHECK YOUR PROGRESS

Check Your Progress 1

Ans to Q1: Calvin Mooers coined the term "Information Retrieval" in 1950.

Ans to Q2: (a) – (ii); (b) – (i); (c) – (iii).

Ans to Q3: a. Natural

Ans to Q4: Any two from: web-based IR systems, digital libraries, multimedia IR systems, distributed IR systems.

Ans to Q5: a. False. IR systems handle unstructured or semi-structured information, while DBMS handles structured data; they serve distinct functions.

Check Your Progress 2

Ans to Q1: Chain Indexing System.

Ans to Q2: a. Universal repository.

Ans to Q3: (a) – (ii); (b) – (i); (c) – (iii).

Ans to Q4: a. False. Traditional systems lack the capacity to handle semi-structured or unstructured data types, diverse file formats, or the scale of modern digital information.

Check Your Progress 3

Ans to Q1: 1978.

Ans to Q2: (a) – (ii); (b) – (i); (c) – (iii); (d) – (iv).

Ans to Q3: a. Polytechnic.

Ans to Q4: ASK stands for Anomalous State of Knowledge.

Ans to Q5: a. False. Calvin Mooers coined the term "Information Retrieval" in 1950.

1.8 MODEL QUESTIONS

Short Answer Questions

1. Define Information Retrieval in your own words, citing any one formal definition.
2. What is the difference between an IR system and a Database Management System?
3. Name any three features of an Information Retrieval system.
4. What is meant by "precision" in the context of IR?
5. What is "recall" in information retrieval? How does it differ from precision?
6. Who is credited with coining the term "Information Retrieval," and in what year?
7. What was the significance of Vannevar Bush's Memex system?
8. State any two reasons that necessitated the development of Information Retrieval as a field.
9. What was the SMART system, and who developed it?
10. In what year was the World Wide Web proposed, and by whom?

Long Answer Questions

1. Discuss in detail the definition, features, scope, and objectives of Information Retrieval. How has the scope of IR evolved over time?
2. Explain the various factors that necessitated the development of Information Retrieval. How did the limitations of traditional library systems contribute to the emergence of IR?
3. Trace the historical development of Information Retrieval from early cataloguing practices to the introduction of computers. Discuss the contributions of Emanuel Goldberg, Vannevar Bush, and Gerard Salton to the field.
4. Write a detailed note on the key milestones in the history of Information Retrieval from 1950 to 2003, with reference to major systems, conferences, and contributions.

1.9 REFERENCES AND FURTHER READINGS

References

- Kowalski, G.J., and Maybury, M.T. *Information Storage and Retrieval Systems: Theory and Implementation* (2nd/3rd ed.). Kluwer Academic Publishers. https://theswissbay.ch/pdf/Gentoomen%20Library/Information%20Retrieval/Information%20Storage%20And%20Retrieval%20Systems-Theory%20And%20Impl%202e_Kowalski%20GJ%20%282002%29.pdf
- Meadow, C.T., Boyce, B.R., Kraft, D.H., and Barry, C.L. *Text Information Retrieval Systems* (3rd ed.). Academic Press, 2007. ISBN: 0-12-369412-4. https://diglibrary.weebly.com/uploads/1/8/5/1/18511482/text_info_retrieval_system.pdf
- Chu, H. *Information Representation and Retrieval in the Digital Age*. Medford, NJ: Information Today Press, 2003. (ASIST Monograph Series.) 220479751_Information_Representation_and_Retrieval_in_the_Digital_Age_ASIST_Monograph_Series_by_Heting_Chu_Medford_NJ_Information_Today_2003_248_p_ISBN_1
- Salton, G., Yang, C-S., and Yu, C.T. "A Theory of Term Importance in Automatic Text Analysis." *Journal of the American Society for Information Science*, 26(1), 1975, pp. 33-44.
- Salton, G., Wong, A., and Yang, C-S. "A Vector Space Model for Automatic Indexing." *Communications of the ACM*, 18(11), 1975, pp. 613-620.
- Mitev, N.N., Venner, G.M., and Walker, S. *Designing an Online Public Access Catalogue: Okapi, a Catalogue on a Local Area Network*. The British Library, 1985.
- Sammon, J.W. "A Nonlinear Mapping for Data Structure Analysis." *IEEE Transactions on Computers*, 18(5), 1969, pp. 401-409.

Further Readings

- Manning, C.D., Raghavan, P., and Schutze, H. *Introduction to Information Retrieval*. Cambridge University Press, 2008.
- Baeza-Yates, R., and Ribeiro-Neto, B. *Modern Information Retrieval: The Concepts and Technology behind Search* (2nd ed.). Addison-Wesley, 2011.
- Sparck Jones, K., and Willett, P. (eds.). *Readings in Information Retrieval*. Morgan Kaufmann, 1997.

*** ***** ***

UNIT 2: BASIC CONCEPTS AND COMPONENTS OF IR SYSTEMS

UNIT STRUCTURE

2.1 Learning Objectives
2.2 Introduction
2.3 Types of IR Systems
2.4 Basic Components of an IR System
    2.4.1 Indexing: Creating Document Representations
    2.4.2 Query Formulation: Creating Query Representations
    2.4.3 Matching Query and Document Representations
    2.4.4 Selection
    2.4.5 Relevance Feedback and Interactive Retrieval
2.5 Purpose and Functions of IR Systems
    2.5.1 Purpose
    2.5.2 Functions
2.6 Let Us Sum Up
2.7 Answer to Check Your Progress
2.8 Model Questions
2.9 References and Further Readings

2.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- Identify the major types of Information Retrieval (IR) systems
- Describe the basic components involved in the IR process
- Explain the purpose and core functions of IR systems

2.2 INTRODUCTION

The rapid expansion of information across digital, print, and multimedia formats has made systematic methods of storing and retrieving information indispensable. An Information Retrieval (IR) system provides the technological infrastructure and procedural framework that enables users to locate relevant documents or data from large, organised collections.

According to Chowdhury (2010), "the objective of an information retrieval system is to enable users to find relevant information from an organised collection of documents." In its simplest conceptualisation, an IR system maps user information

needs, expressed as queries, against a structured or indexed document collection and returns items that are likely to be relevant. While IR systems originally concentrated on textual documents, contemporary systems deal with diverse media types including audio, video, images, and structured data.

It is also important to distinguish IR systems from conventional database management systems. Unlike a conventional database management system, an information retrieval system deals with unstructured data. The IR system does not merely match records against fixed attributes; it predicts the relevance of documents to a user's information need, which may be imprecise or ambiguous.

2.3 TYPES OF IR SYSTEMS

IR research has extended considerably beyond its initial focus on textual retrieval. Today, a range of specialised IR systems exists, each designed to address a specific type of information and specific user demands.

2.3.1 Text Retrieval Systems

Text retrieval systems are the most widely studied type of IR system and form the foundation of the field. According to Panigrahi, "IR has concentrated more on finding the documents consisting of written text; much IR research focuses more specifically on text retrieval, the computerised retrieval of machine-readable text without human indexing." These systems work with full-text or bibliographic databases, accepting keyword or natural language queries and returning ranked or Boolean-matched results. Modern search engines such as Google are built on principles developed through text retrieval research.

2.3.2 Speech Retrieval Systems

Speech retrieval systems extract and retrieve information from spoken content. According to Panigrahi, "speech is an information-rich element of multimedia" and several well-established research fields such as "speech recognition, speaker identification, voice detection, sentiment analysis and fingerprinting" contribute techniques and methods to speech retrieval. These systems are particularly significant in contexts where audio recordings constitute the primary information source.

2.3.3 Image Retrieval Systems

Image retrieval is a sub-field of IR that focuses on browsing, searching, and retrieving images from large databases. The most popular techniques "utilise some method of adding metadata such as use of captioning, keywords, or descriptions to the images so that retrieval can be performed over the annotation words." The growth of social networks and the shift from the web to the data web has led to the development of several web-based image annotation tools and automated image detection methods.

2.3.4 Question-Answering Systems

Question-answering (QA) systems represent a specialised application within IR that intersects with Natural Language Processing (NLP). A QA system "is involved with building systems that automatically answer questions posed by humans in a natural language." Such systems may draw answers from structured knowledge bases or from unstructured document collections, depending on their design.

2.3.5 Cross-Language Information Retrieval

Cross-language IR (CLIR) "deals with fetching information written in a particular language different from the language of the user's query," such as using Hindi queries to retrieve English documents. This is considered one of the most challenging sub-fields of IR, and significant research efforts have been directed towards developing effective translation and retrieval models for multilingual contexts.

2.3.6 Music Information Retrieval

Music Information Retrieval (MIR) is described as "the interdisciplinary field of retrieving useful information from music." Researchers in this area come from diverse disciplines including computer science, musicology, psychology, signal processing, and machine learning. Extractable features in MIR include frequency of occurrence of notes or chords, harmonies, melody, main pitch, beats per minute, and rhythm.

2.3.7 ISAR Systems: A Broader Typology

The wider concept of Information Storage and Retrieval (ISAR) systems, which encompasses five major types. According to the ISAR framework, these types include: the Database Management System (DBMS), the Text Retrieval System, the Management Information System (MIS), the Decision Support System (DSS), and the Knowledge-Based System (KBS).

- **DBMS:** A DBMS "is primarily concerned with data storage, maintenance and retrieval and is used to keep control and manipulate data within the database." It manages structured data, where files are described by a pre-specified set of attributes.
- **Text Retrieval System:** "In contrast to DBMS, text retrieval systems are designed for unstructured data such as full-text documents." Queries are usually language-based here, such as keywords, and advanced search techniques such as proximity search can be applied.
- **MIS:** A Management Information System is "designed to cope up with the needs of managements who need to have information about different alternatives related to his/her interest to facilitate his work." It is built on a DBMS platform but subjects' information to special processing for managerial decision-making.

- **DSS:** Decision Support Systems "help top-level management in arriving at decisions" by providing possible alternative solutions. They are interactive computer-based systems that "provide the user easy access to decision models and data in order to support semi-structured and unstructured decisions."
- **KBS:** Knowledge-Based Systems are "specialised computer programs, modelled in the same way as human experts tackle problems and arrive at solutions." These systems "rely upon a store of specialised knowledge for solving problems" and are capable of learning from experience through an inferencing mechanism.

Check Your Progress 1

Q1. What does the term "Cross-Language Information Retrieval" mean? **Q2.** Name any three types of ISAR systems.

Q3. Match the following:

Column A	Column B
(i) DBMS	(a) Answers questions in natural language
(ii)QA System	(b) Deals with structured data and pre-specified attributes
(iii) MIR	(c) Retrieves useful information from music
(iv) CLIR	(d) Uses queries in one language to retrieve documents in another

2.4 BASIC COMPONENTS OF AN IR SYSTEM

An IR system performs its retrieval operation through a sequence of tightly interlinked sub-processes. "An IR system performs retrieval operation by indexing documents and designing queries, thereby leading to representation of documents and representation of queries, respectively; the system then matches the indexed documents with that of user query and displays the matched documents found, and the user selects the relevant items." These operations are directly dependent on each other and often proceed iteratively.

2.4.1 Indexing: Creating Document Representations

Indexing is the foundational step through which an IR system organises its document collection for efficient retrieval. Indexing is also referred to as "cataloguing, metadata assignment, or metadata extraction" and is defined as "the manual or automated process of creating indexes for record collections."

More specifically, subject indexing is defined as "the act of describing or classifying a document by index terms or other symbols in order to indicate what the document is about, to summarise its content or to increase its findability." Indexing may be document-oriented, in which case the indexer captures what the document is about, or request-oriented, in which case the indexer assesses the document's relevance to user interests.

Automatic indexing begins with feature selection and extraction, involving the extraction of all words from a text, followed by the elimination of stop-words, stemming, counting most frequent words, and mapping to concepts using a thesaurus or ontology. For images, extractable features include colour distribution, texture, and shape detection. For music, extractable features comprise frequency of notes, harmonies, melody, and rhythm.

The ISAR framework further specifies two primary methods of knowledge representation within the index or knowledge base: semantic networks, frame-based representation, and rule-based representation. A semantic network "contains points called nodes connected by links called arcs," where nodes represent objects or concepts and arcs represent relations between them. In rule-based representation, "rules are expressed in the form of IF-THEN statements" to specify how inferences are to be drawn from stored data.

2.4.2 Query Formulation: Creating Query Representations

Query formulation is the process by which a user's information need is transformed into a formal representation that can be matched against the indexed collection. According to Panigrahi, "a user's query can be transformed, manually or automatically, into a formal query representation, also called query formulation."

A query may take various forms. It may be a free-text search, involving words or phrases that the system should acknowledge, or a structured query using Boolean operators (AND, OR, NOT). A Boolean query "specifies three conditions": AND narrows the search and retrieves records containing all of the specified terms; OR broadens the search and retrieves records containing any of the terms; NOT narrows the search by excluding records containing a specified term.

Advanced search interfaces may also support proximity search, truncation, and controlled vocabulary searches. The ISAR framework notes that "in advanced search, Boolean operators, like AND, OR, NOT, etc., can be incorporated to refine a query." The system may also use thesauri and synonym expansion to suggest narrower, broader, and related terms, assisting users who may not know the precise vocabulary of the subject domain.

2.4.3 Matching Query and Document Representations

The matching phase is where the formal query representation is compared against document representations in the index. "The document relevance is predicted by identifying the relevant features of the query with that of the document."

In exact match systems, the system marks documents that satisfy all the conditions of a Boolean query, predicting relevance as either 1 (relevant) or 0 (not relevant). In ranked retrieval systems, documents are assigned a score of expected relevance and returned in ranked order. In this model, each term's contribution to the relevance score is calculated as a product of the term's weight in the query, the term frequency (tf) in the document, and the inverse document frequency (idf), which measures the rarity of the term across the entire collection.

The key performance measures used to evaluate this matching process are recall and precision. Recall is defined as "the number of relevant documents retrieved divided by the total number of relevant documents in the collection," and precision is defined as "the number of relevant documents retrieved divided by the total number of documents retrieved." These two measures have an inverse relationship: increasing recall often reduces precision and vice versa.

2.4.4 Selection

Following retrieval, the user examines the results and selects the documents most relevant to the information need. "Results can be organised in rank order; the search process can be stopped once the user's need is fulfilled; in case of groupings, the documents based on subject, automatic classification scheme or clustering techniques similar items can be examined side by side."

The display of results plays an important role in effective selection. "The display of titles along with the abstract with key terms highlighted is considered to be the most useful, as title alone is too short, the full text too long." Clustering algorithms, which group similar documents together, further assist users in navigating large result sets. According to the ISAR framework, "a clustering algorithm attempts to identify natural groups of components or data based on some similarity in a given population."

2.4.5 Relevance Feedback and Interactive Retrieval

Relevance feedback allows the IR system to improve retrieval performance iteratively, based on user assessment of initial results. "Once the user has evaluated the significance of a few items found, the query can be made better." The system provides assistance by displaying features, such as assigned descriptors and text words, found in many relevant items and another list from irrelevant items.

In automatic relevance feedback, "the system can automatically improve the query by identifying those unique features which can distinguish between relevant from irrelevant items and thus are good predictors of relevance." This mechanism was formalised through the work of Gerard Salton and colleagues at Cornell University, whose SMART system incorporated "many important concepts like vector space model, relevance feedback, and Rocchio Classification."

Check Your Progress 2

Q1. What is "stemming" in the context of automatic indexing?

Q2. Fill in the blank:

a. The _____operator in Boolean retrieval broadens the search and retrieves records containing any of the specified terms.

Q3. Which two performance measures are most commonly used to evaluate an IR system's matching effectiveness?

Q4. True or False: In exact-match retrieval, the system assigns a ranked relevance score to every document.

Q5. Name the component of an ISAR system responsible for taking input from the search interface and searching the knowledge base.

2.5 PURPOSE AND FUNCTIONS OF IR SYSTEMS

2.5.1 Purpose

The primary purpose of an IR system is to bridge the gap between information creators and information users. According to Panigrahi, "an information retrieval system functions as a bridge between the world of creators or generators of information and the users who are ignorant of that information." In other words, the system ensures that a piece of knowledge created by one individual and recorded in a document can be discovered and accessed by another individual who needs that knowledge but is unaware of its existence.

The main aim, is "to collect and organise information in one or more fields in order to help the users to access the retrieved resources." The ISAR framework similarly states that "the principle objective of ISAR system is to provide correct information to the user in least time with least efforts."

The concept of relevance is central to the purpose of IR systems. All operations in IR "surround around usefulness and relevance of documents." A resource is considered relevant if it contains information that either directly answers the user's query or can be used in combination with other information to infer an answer or perform a task. Relevance is a matter of degree; some documents are highly relevant and indispensable, while others may contribute marginally to the user's requirements.

The ISAR framework also emphasises that the system should function as a non-ambiguous facilitator: "the system should be so organised that ambiguity of information is avoided so that search result is free from any kind of ambiguity." This requires careful identification of terms, setting of context, and proper indexing.

2.5.2 Functions

The functions of an IR system describe the operational steps through which its purpose is realised. Drawing from Chowdhury (2010) as cited in Panigrahi, the different functions of IR systems can be enumerated as follows:

1. **Identifying information sources:** To identify the information sources relevant to the areas of interest of the target user community. This is particularly challenging in the web environment, where a very large and heterogeneous body of content must be monitored and evaluated.
2. **Analysing the contents of sources:** To analyse the contents of documents, which is "becoming increasingly challenging as the size, volume and variety of information sources is increasing rapidly; web information retrieval is carried out automatically using specially designed programmes called spiders."

3. **Representing document contents:** To represent the contents of analysed sources in a way that matches users' queries, achieved "by automatically creating one or more index files."
4. **Analysing and representing user queries:** To analyse users' queries and represent them in a form suitable for matching the database. This is done through the design of sophisticated search interfaces "including those that can provide some help to users for selection of appropriate search terms by using dictionary and thesauri, automatic spell checkers, a predefined set of search statements."
5. **Matching queries with the database:** To match the search statement with the stored database. "A number of complex information retrieval models have been developed over the years that are used to determine the similarity of the query and stored documents."
6. **Retrieving relevant information:** To retrieve relevant information using "a variety of tools and techniques to determine the relevance of retrieved items and their ranking."
7. **Continuous system improvement:** To make continuous changes in all aspects of the system, taking into account "the rapid developments in information and communication technologies (ICTs) relating to changing patterns of society, users, and their information needs and expectations."

In addition to the above, the ISAR framework specifies that an effective IR system must support large databases and "retrieve relevant documents in response to queries interactively, often within 1 to 10 seconds." The system must also enable users to create, insert, modify, and delete documents in the database, and serve as a platform for users to search and examine retrieved documents.

Check Your Progress 3

Q1. What are "spiders" in the context of web-based IR systems?

Q2. True or False:

a. Relevance in information retrieval can only be measured in binary terms.

2.6 LET US SUM UP

We have learned the basic concepts and components of Information Retrieval systems. It began by mapping out the principal types of IR systems, ranging from text retrieval systems to speech, image, question-answering, cross-language, and music information retrieval systems, while also contextualising these within the broader ISAR typology that includes DBMS, MIS, DSS, and KBS. The five core components of the IR process: indexing, query formulation, matching, selection, and relevance feedback, each of which plays a distinct and interrelated role in ensuring that users receive relevant results. The purpose of IR systems is bridging the gap between information

creators and users, and detailed the operational functions through which this purpose is achieved. The dual measures of recall and precision serve as the primary criteria for evaluating how well an IR system fulfils these functions.

2.7ANSWER TO CHECK YOUR PROGRESS

Check Your Progress 1

Ans to Q1: Cross-Language Information Retrieval is an area of IR that deals with fetching information written in a language different from the language used in the user's query, such as using a Hindi query to retrieve English documents.

Ans to Q2: Three types of ISAR systems are: (i) Database Management System (DBMS), (ii) Text Retrieval System, and (iii) Knowledge-Based System (KBS).

Ans to Q3: (i) - (b); (ii) - (a); (iii) - (c); (iv) - (d).

Check Your Progress 2

Ans to Q1: Stemming is defined as "the process for reducing inflected words to their stem, base or root form," so that related words map to the same stem even if that stem is not itself a valid dictionary root.

Ans to Q2: OR.

Ans to Q3: Recall and Precision.

Ans to Q4: False. In exact-match retrieval, relevance is predicted as 1 or 0 (present or absent). Ranked scoring is a feature of ranked retrieval systems.

Ans to Q5: The Search Agent.

Check Your Progress 3

Ans to Q1: Spiders are "specially designed programmes" used in web-based IR to automatically crawl and analyse the contents of web documents. They go through various web pages on the Internet and index them word-by-word; this index is stored inside databases similar to library catalogues.

Ans to Q2: False. Relevance "cannot be measured only by yes/no terms, but also by the degree of relevancy."

2.8 MODEL QUESTIONS

Short Questions

1. Define Information Retrieval in your own words, drawing from the definitions discussed in this unit.

2. What is the difference between a Text Retrieval System and a Database Management System (DBMS)?
3. Name and briefly describe any two types of IR systems discussed in this unit.
4. What is an "inverted index" and why is it important in IR?
5. Differentiate between recall and precision as measures of retrieval performance.
6. What is relevance feedback, and how does it improve IR system performance?
7. What is the role of Boolean operators in query formulation?
8. List any three functional objectives of an IR system as described in this unit.

Long Questions

1. Discuss in detail the basic components involved in the information retrieval process. How are these components interdependent?
2. Explain the different types of IR systems with examples. How do the needs of users determine the type of system to be designed?
3. Examine the purpose and functions of an Information Retrieval system. How does the concept of relevance underpin these functions?
4. Write a detailed note on the process of indexing in IR systems, covering both manual and automatic approaches.
5. Critically examine the relationship between recall and precision in IR systems. Why is it difficult to achieve a high degree of both simultaneously?

2.9 REFERENCES AND FURTHER READINGS

References

- Chowdhury, G. G. (2010). *Introduction to Modern Information Retrieval* (3rd ed.). London: Facet Publishing.
- Lancaster, F. W. (1968). *Information Retrieval Systems: Characteristics, Testing and Evaluation*. New York: John Wiley.
- Lancaster, F. W. (1979). *Information Retrieval Systems: Characteristics, Testing and Evaluation* (2nd ed.). New York: John Wiley.
- Meadow, C. T., Boyce, B. R., Kraft, D. H., and Barry, C. (2007). *Text Information Retrieval Systems* (3rd ed.). London: Academic Press.
- Panigrahi, P. K. (n.d.). Basic Concepts and Components of Information Retrieval Systems. Module 2, e-PG Pathshala. https://epgp.inflibnet.ac.in/epgpdata/uploads/epgp_content/S000021LI/P000207/M004284/ET/1483076127P07_M02.pdf
- Salton, G. and McGill, M. (1983). *Introduction to Modern Information Retrieval*. New York: McGraw-Hill.

- Sparck Jones, K. and Willett, P. (eds.) (1997). *Readings in Information Retrieval*. San Francisco: Morgan Kaufmann.
- Vickery, B. C. (1970). *Techniques of Information Retrieval*. London: Butterworth.
- Vickery, B. and Vickery, A. (1987). *Information Science in Theory and Practice*. London: Bowker-Saur.
- Block-4-2 Course Material. (n.d.). *Information Storage and Retrieval Systems: Objectives, Types, Operations and Design*. New Delhi: Indira Gandhi National Open University. http://egyankosh.ac.in//handle/123456789/76438

Further Readings

- Baeza-Yates, R. and Ribeiro-Neto, B. (2011). *Modern Information Retrieval: The Concepts and Technology behind Search* (2nd ed.). Harlow: Addison-Wesley.
- Belkin, N. J. (1980). Anomalous states of knowledge as a basis for information retrieval. *Canadian Journal of Information Science*, 5, 133–143.
- Hearst, M. (2009). *Search User Interfaces*. Cambridge: Cambridge University Press.
- Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge: Cambridge University Press. Available at: https://nlp.stanford.edu/IR-book/
- Rowley, J. (1996). *The Basics of Information Systems* (2nd ed.). London: Library Association Publishing.
- Salton, G. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11), 613–620.

*** ***** ***

UNIT 3: DATABASE MANAGEMENT SYSTEMS

UNIT STRUCTURE

3.1 Learning Objectives
3.2 Introduction to Database Management Systems
3.3 Advantages of DBMS in the ISAR Context
  3.3.1 Data Independence
  3.3.2 Reduction of Data Redundancy and Inconsistency
  3.3.3 Efficient Data Access and Retrieval
  3.3.4 Data Integrity and Security
  3.3.5 Concurrent Access and Multi-User Support
  3.3.6 Backup and Recovery
  3.3.7 Application to Online Public Access Catalogues (OPACs)
3.4 ACID Properties and Their Relevance to IR Databases
  3.4.1 Atomicity
  3.4.2 Consistency
  3.4.3 Isolation
  3.4.4 Durability
  3.4.5 ACID Properties in the Context of IR Databases
3.5 Data Models in DBMS
  3.5.1 Object-Based Logical Model
  3.5.2 Record-Based Logical Model
3.6 DBMS Languages
  3.6.1 Data Definition Language (DDL)
  3.6.2 Data Manipulation Language (DML)
  3.6.3 Data Control Language (DCL)
3.7 Let Us Sum Up
3.8 Answer to Check Your Progress
3.9 Model Questions
3.10 References and Further Readings

3.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- Define a Database Management System and explain its purpose in information environments
- Describe the key advantages of DBMS within the context of Information Storage and Retrieval

- Explain the ACID properties and their relevance to the maintenance of IR databases
- Distinguish between object-based and record-based logical data models
- Identify and explain the principal DBMS languages: DDL, DML, and DCL

3.2 INTRODUCTION TO DATABASE MANAGEMENT SYSTEMS

The systematic organisation, storage, and retrieval of information constitute the foundational concern of library and information science. As the volume and complexity of information have grown substantially in the digital era, manual or file-based systems have proven inadequate for managing large bodies of structured data. It is in this context that the Database Management System (DBMS) has emerged as an indispensable tool for modern information professionals.

According to Silberschatz, Korth, and Sudarshan in *Database System Concepts* (2002), "a Database Management System (DBMS) is a collection of interrelated data and a set of programs to access those data. The primary goal of a DBMS is to provide a way to store and retrieve database information that is both convenient and efficient." This definition captures two fundamental dimensions: the organisational aspect of data storage and the operational aspect of data retrieval.

A database, in the formal sense, is a structured collection of data representing some aspect of the real world, often referred to as the "mini-world" or the "Universe of Discourse (UoD)." A DBMS contains information relevant to a specific enterprise or domain and provides an environment designed to be both convenient for users and efficient in its operations. In the context of library and information science, the "enterprise" may be a library, an archive, a documentation centre, or any information system that must store, organise, and retrieve information objects.

Before the advent of DBMS technology, information was managed through flat-file systems in which each application maintained its own set of data files. This approach led to significant problems, including data redundancy, inconsistency, difficulty in accessing data, integrity problems, atomicity failures, and limited support for concurrent access. The DBMS was developed to address precisely these challenges by providing centralised data management with a shared and controlled infrastructure.

The components of a DBMS typically include a query processor, a storage manager, a transaction manager, a concurrency control manager, and a data dictionary. Together, these components allow users and applications to interact with the database through well-defined interfaces, abstracting the complexity of physical data storage.

3.3 ADVANTAGES OF DBMS IN THE ISAR CONTEXT

Information Storage and Retrieval (ISAR) systems are designed to facilitate the recording, organisation, and retrieval of information. Traditional file-based approaches in libraries and documentation centres were associated with significant inefficiencies.

The adoption of DBMS in ISAR environments has provided a range of functional, administrative, and technical benefits.

3.3.1 Data Independence

According to the INFLIBNET eBook on *Information Storage and Retrieval*, "the DBMS can provide an abstract view of the data insulated from application codes from details of data representation and storage." In ISAR systems, this means that changes to the physical organisation of bibliographic data (such as changes to storage formats or indexing schemes) do not require modifications to the application programs used by cataloguers or patrons. This separation between the logical and physical layers is called data independence and is one of the most practically significant advantages of DBMS.

3.3.2 Reduction of Data Redundancy and Inconsistency

Traditional file systems in libraries often resulted in the same bibliographic record being duplicated across multiple departments or applications, leading to inconsistency when updates were applied in one location but not another. DBMS addresses this by storing data in a centralised and structured manner, minimising duplication. It supports shared access across applications, eliminating repeated storage and ensuring that data remains consistent across all views of the system.

3.3.3 Efficient Data Access and Retrieval

According to the INFLIBNET, "DBMS utilises a variety of sophisticated techniques to store and retrieve data efficiently. This feature is especially important if data is stored on external storage devices." In ISAR systems, where collections may contain millions of records, the ability to use indexing, query optimisation, and structured query languages makes retrieval considerably faster and more precise than any manual or flat-file approach. Complex queries based on multiple criteria (author, subject, date, classification number) can be executed rapidly.

3.3.4 Data Integrity and Security

DBMS enforces integrity constraints on data, ensuring that only valid data is entered or modified. In an ISAR context, this may include ensuring that mandatory fields (such as title or ISBN) are present in bibliographic records, or that foreign key relationships between records and patrons remain valid. In addition, DBMS provides access control mechanisms that restrict which users may view, modify, or delete records, protecting the integrity of the information collection.

3.3.5 Concurrent Access and Multi-User Support

In a library or documentation centre, multiple users frequently need to access the same records simultaneously. DBMS handles concurrent data access using concurrency control mechanisms such as locking protocols, timestamp ordering, and optimistic concurrency control. These mechanisms prevent data corruption and ensure that each user's interaction with the system is consistent and accurate.

3.3.6 Backup and Recovery

DBMS systems provide automated backup and point-in-time recovery capabilities. In ISAR environments, where the loss of bibliographic or transaction data could be highly disruptive, this feature ensures that the system can be restored to a consistent state following hardware failure, software errors, or other disruptions.

3.3.7 Application to Online Public Access Catalogues (OPACs)

A notable application of DBMS in the ISAR domain is the Online Public Access Catalogue (OPAC). "Tools like Database Management System (DBMS) are used for keeping records of a holding of library, which is known as Online Public Access Catalogue (OPAC)." In ISAR Systems, DBMS form the technological backbone of modern library management systems, enabling functions such as cataloguing, circulation management, and serial control.

Check Your Progress 1

Q1. What does DBMS stand for?

Q2. Match the following:

Column A	Column B
(a) Data Independence	(i) Prevention of duplicate records
(b) Data Redundancy	(ii) Automated restoration after failure
(c) Backup and Recovery	(iii) Logical structure separated from physical storage

Q3. Fill in the blank: The full form of OPAC is_____ .

Q4. According to Silberschatz, Korth, and Sudarshan, what is the primary goal of a DBMS?

3.4 ACID PROPERTIES AND THEIR RELEVANCE TO IR DATABASES

A central concern in any database system is ensuring that the data remains accurate and reliable even in the presence of failures, errors, or simultaneous access by multiple users. This reliability is formally guaranteed by the ACID properties, which define the behavioural requirements of database transactions.

A transaction is defined as a collection of operations that performs a single logical function in a database application. "ACID properties safeguard the data integrity of a DBMS by ensuring that transactions either complete successfully or leave no trace if interrupted. They prevent partial updates from corrupting the data and ensure that the database transitions only between valid states."

ACID is an acronym for four properties: Atomicity, Consistency, Isolation, and Durability.

3.4.1 Atomicity

Atomicity requires that a transaction be treated as a single, indivisible unit of work. According to Silberschatz's, "the DBMS executes either all actions of the transaction or none. Incomplete transactions should not result in permanent changes." In an IR context, consider a transaction in which a new bibliographic record is added and simultaneously the subject index is updated. Atomicity ensures that either both operations complete successfully, or neither takes effect, preventing a situation in which the record exists in the database without a corresponding index entry.

3.4.2 Consistency

Consistency requires that a transaction bring the database from one valid state to another valid state, respecting all defined integrity constraints. In an ISAR system, consistency constraints may include rules such as "every circulation record must have a valid patron ID" or "no two records may share the same accession number." If a transaction would violate such a constraint, the DBMS rolls it back, preserving the internal coherence of the database.

3.4.3 Isolation

Isolation ensures that concurrent transactions are executed in a manner that produces results equivalent to their serial execution. This means that the intermediate states of a transaction are not visible to other concurrently executing transactions. In a library management system, if two staff members are simultaneously updating the same patron record, isolation ensures that their changes do not interfere with each other, preventing lost updates or dirty reads.

3.4.4 Durability

Durability guarantees that once a transaction has been committed, its effects persist even in the event of a subsequent system failure. In practical terms, this means that committed data is written to non-volatile storage and can be recovered following power failures or system crashes. For an ISAR system managing circulation records or acquisition data, durability ensures that completed transactions are not lost due to unforeseen technical failures.

3.4.5 ACID Properties in the Context of IR Databases

The relevance of ACID properties to ISAR systems is substantial. Information retrieval databases store not only bibliographic records but also transactional data such as circulation records, loan histories, acquisition orders, and user access logs. The reliability of these records depends directly on transactional integrity. Atomicity ensures that partial updates do not corrupt the catalogue; consistency ensures that constraint violations are prevented; isolation protects the system from concurrency errors and durability ensures that the results of completed operations are permanently recorded. Together, the ACID properties form the transactional foundation upon which trustworthy information systems are built.

Check Your Progress 2

Q1. What does ACID stand for?

Q2. Fill in the blank: Durability ensures that once a transaction is _, its effects persist even after a system failure.

Q3. Which ACID property ensures that intermediate states of a transaction are not visible to other concurrent transactions?

Q4. Match the following:

Column A	Column B
(a) Atomicity	(i) Database moves from one valid state to another
(b) Consistency	(ii) All or nothing execution
(c) Isolation	(iii) Concurrent transactions do not interfere

Q5. In an ISAR context, give one example of a consistency constraint.

3.5 DATA MODELS IN DBMS

A data model is a conceptual framework used to describe how data is organised, represented, and related within a database system. DBMS resource, "a data model in a DBMS is a conceptual representation of data structures, relationships, and constraints.

It provides a framework for organizing and manipulating data." Data models operate at different levels of abstraction and serve as the bridge between the real-world information environment and the physical storage of data.

DBMS data models are broadly classified into two major categories: object-based logical models and record-based logical models. These two categories differ in the way they represent the structure of data and the relationships among data entities.

3.5.1 Object-Based Logical Model

Object-based logical models describe data in terms of objects, their properties, and the relationships between them. They operate at a high level of abstraction, closely reflecting the conceptual view of the real world. The two most important sub-types are:

Entity-Relationship (ER) Model: The ER model represents the database as a collection of entities (real-world objects or concepts) and the relationships between them. Entities possess attributes, and relationships are depicted through diagrams known as ER diagrams. The ER model is widely used in the initial design phase of DBMS implementation and is particularly useful in the ISAR domain for modelling the relationships between documents, authors, subjects, and patrons.

Object-Oriented Data Model: According to the Sprinkle Data guide on database models, "the object-oriented data model integrates principles from object-oriented programming, such as encapsulation, inheritance, and polymorphism, into database management. Data is stored as objects, which can include both data and methods." Each object has a unique identity independent of the values it contains, and two objects with the same values are considered distinct. This model is particularly suited to applications involving complex, multimedia, or nested data, such as digital library systems handling images, audio, and video documents.

Object-based models are generally used for conceptual and logical design rather than physical implementation and are characterised by their ability to capture semantically rich representations of data.

3.5.2 Record-Based Logical Model

Record-based logical models organise data into fixed-format records, each of which contains a number of fields. These models operate closer to the physical level and are the basis for most conventional database implementations. The resource on record-based data models, the three most widely accepted models under this category are: the Relational Model, the Network Model, and the Hierarchical Model.

Relational Model: The relational model represents data as a set of tables (relations), where each table has multiple columns (attributes) and each row represents a record (tuple). The relational model offers advantages including "structural independence, simplicity, ease of designing, implementation, and ad-hoc query capability." It is the most widely used data model in modern DBMS and forms the basis of SQL-based systems. In the ISAR context, bibliographic databases such as those

underlying integrated library systems are typically implemented using the relational model.

Hierarchical Model: The hierarchical model organises data in a tree-like structure, where each parent record may have one or more child records, but each child record has only one parent. This model is well-suited to data with natural hierarchical relationships, such as a classification sys tem in which main classes contain subclasses, which in turn contain specific subjects. Its disadvantage lies in its rigidity, it cannot efficiently represent many-to-many relationships.

Network Model: The network model extends the hierarchical model by allowing each record to have multiple parent records, thus representing many-to-many relationships. It uses a graph structure in which nodes represent records and edges represent relationships. While more flexible than the hierarchical model, it is more complex to design and navigate, and has largely been superseded by the relational model in practice.

Check Your Progress 3

Q1. Fill in the blank: The relational model represents data in the form of_____ .

Q2. Which data model organises data in a tree-like structure with parent-child relationships?

Q3. Match the following:

Column A	Column B
(a) Relational Model	(i) Graph structure with multiple parent records
(b) Network Model	(ii) Tables with rows and columns
(c) Hierarchical Model	(iii) Tree-like parent-child structure

Q4. State true or false: In an object-oriented data model, two objects containing identical values are considered the same object.

3.6 DBMS LANGUAGES

Interaction with a DBMS is mediated through specialised languages that allow users and administrators to define, manipulate, and control the data within the database. The database languages resource, "database languages are specialised languages used to interact with a database. They allow users to perform different tasks such as defining, controlling, and manipulating the data." The primary languages used in DBMS are the Data Definition Language (DDL), the Data Manipulation Language (DML), and the Data Control Language (DCL).

3.6.1 Data Definition Language (DDL)

The Data Definition Language is used to define and modify the structure of the database, including its schema, tables, indices, and constraints. "Data Definition Language (DDL) is the database language that is used to create, delete, or modify database schema." DDL commands operate at the level of the database structure rather than its content.

The principal DDL commands include:

- **CREATE:** Used to create a new database, table, view, or index. For example, a cataloguer implementing a new bibliographic database would use CREATE TABLE to define the fields for author, title, subject, and publication year.
- **ALTER:** Used to modify an existing database object, such as adding a new column to an existing table.
- **DROP:** Used to permanently delete a table, view, or other database object from the schema.
- **TRUNCATE:** Used to remove all records from a table while retaining the table structure.

In the ISAR context, DDL commands are employed during the design and maintenance phases of a library management system, for example when creating a new table to store digital object metadata or when altering an existing catalogue schema to accommodate new fields required by an updated cataloguing standard such as RDA (Resource Description and Access).

3.6.2 Data Manipulation Language (DML)

The Data Manipulation Language is used to interact with the data stored within the database. It enables users to insert new records, query existing records, update records, and delete records. DML operations constitute the most frequent type of database interaction in an operational ISAR system.

The principal DML commands include:

- **SELECT:** Used to retrieve data from one or more tables based on specified criteria. This is the most frequently used command in information retrieval operations.
- **INSERT:** Used to add new records to a table, such as when a new bibliographic item is added to the catalogue.
- **UPDATE:** Used to modify the values in existing records, such as when correcting a bibliographic error or updating a patron's contact information.
- **DELETE:** Used to remove specific records from a table.

In an ISAR system, DML is central to everyday operations. A reference librarian performing a subject search, a circulation staff member recording the return of a book, or an acquisitions officer updating the status of an order is, in each case, interacting

with the database through DML operations. The SELECT command in particular is fundamental to all retrieval operations and supports the use of complex logical operators, wildcards, and Boolean expressions that form the basis of structured information retrieval.

3.6.3 Data Control Language (DCL)

The Data Control Language is used to manage access rights and permissions within the database. In a multi-user information environment such as a library network, it is essential that different categories of users have different levels of access to the database.

The principal DCL commands are:

- **GRANT:** Used to assign specific privileges to a user or group of users. For example, a GRANT command might allow library patrons to perform SELECT operations on the catalogue while restricting INSERT or DELETE operations to cataloguing staff.
- **REVOKE:** Used to withdraw previously granted privileges from a user.

DCL is directly relevant to data security in ISAR systems. By controlling who may read, write, modify, or delete records, DCL commands implement the access control policies that protect the integrity and confidentiality of the information collection. In practice, a database administrator in a library system uses DCL to assign roles and permissions to cataloguers, circulation staff, systems administrators, and public users, each with an appropriately defined set of privileges.

Check Your Progress 4

Q1. What is the full form of DDL, DML, and DCL?

Q2. Match the following commands with their language category:

Column A	Column B
(a) SELECT	(i) DDL
(b) CREATE	(ii) DML
(c) GRANT	(iii) DCL

Q3. Fill in the blank: The command is used to remove previously assigned database privileges from a user.

Q4. Which DBMS language is primarily concerned with data security and access control?

Q5. Name the DDL command used to permanently delete a table from the database schema.

3.7 LET US SUM UP

We have learned the definition of DBMS and its distinction from traditional file-based systems, proceeded to examine the specific advantages that DBMS offers in ISAR environments, and then explored the ACID properties that ensure transactional reliability. The two major categories of data models include object-based and record-based. The three principal DBMS languages are DDL, DML, and DCL. Each of these components is interconnected: an effective ISAR system requires a well-designed data model, implemented using DDL, operated through DML for retrieval and manipulation, secured through DCL for access control, and maintained with transactional integrity through ACID compliance.

3.8 ANSWER TO CHECK YOUR PROGRSS

Check Your Progress 1

Ans to Q1. DBMS stands for Database Management System.

Ans to Q2. (a) - (iii); (b) - (i); (c) - (ii).

Ans to Q3. Online Public Access Catalogue.

Ans to Q4. According to Silberschatz, Korth, and Sudarshan, the primary goal of a DBMS is to provide a way to store and retrieve database information that is both convenient and efficient.

Check Your Progress 2

Ans to Q1. ACID stands for Atomicity, Consistency, Isolation, and Durability.

Ans to Q2. Committed.

Ans to Q3. Isolation.

Ans to Q4. (a) - (ii); (b) - (i); (c) - (iii).

Check Your Progress 3

Ans to Q1. Tables (relations).

Ans to Q2. The Hierarchical Model.

Ans to Q3. (a) - (ii); (b) - (i); (c) - (iii).

Ans to Q4. False. Each object has its own unique identity and is distinct from other objects even if they contain the same values.

Check Your Progress 4

Ans to Q1. DDL: Data Definition Language; DML: Data Manipulation Language; DCL: Data Control Language.

Ans to Q2. (a) - (ii); (b) - (i); (c) - (iii).

Ans to Q3. REVOKE.

Ans to Q4. Data Control Language (DCL).

Ans to Q5. DROP.

3.9 MODEL QUESTIONS

Short Questions

1. Define a Database Management System. Whose definition is commonly cited in this context?
2. What is meant by data independence in DBMS?
3. Distinguish between Atomicity and Consistency in the context of ACID properties.
4. What is the role of DDL in a DBMS?
5. Name the three models under the record-based logical data model category.
6. What is the function of the REVOKE command in DCL?
7. How is the OPAC related to DBMS in library management?
8. What is a transaction in database terminology?
9. Differentiate between the hierarchical model and the network model.
10. What is the SELECT command used for in DML?

Long Questions

1. Describe the advantages of DBMS over traditional file-based systems, with specific reference to the ISAR context.
2. Explain the four ACID properties of a DBMS. Discuss their relevance to the maintenance of reliable information retrieval databases.
3. Write a detailed note on DBMS languages (DDL, DML, DCL), explaining the purpose and principal commands of each language. How are these languages used in the management of an integrated library system?
4. Discuss the role of DBMS in modern Information Storage and Retrieval systems. How has DBMS technology transformed the management of library catalogues and documentation systems?

3.10 REFERENCES AND FURTHER READING

References

- Silberschatz, A., Korth, H. F., and Sudarshan, S. (2002). *Database System Concepts* (4th ed.). McGraw-Hill.
- GeeksforGeeks. (2026). *ACID Properties in DBMS*. Retrieved from https://www.geeksforgeeks.org/dbms/acid-properties-in-dbms/
- GeeksforGeeks. (2026). *Database Languages in DBMS*. Retrieved from https://www.geeksforgeeks.org/dbms/database-languages-in-dbms/
- GeeksforGeeks. (2024). *Record-Based Data Model*. Retrieved from https://www.geeksforgeeks.org/dbms/record-based-data-model/
- GeeksforGeeks. (2025). *Advantages of Database Management System*. Retrieved from https://www.geeksforgeeks.org/dbms/advantages-of-database-management-system/
- INFLIBNET Centre. (n.d.). *Database Management System*. In *Information Storage and Retrieval* (eBook). Retrieved from https://ebooks.inflibnet.ac.in/lisp7/chapter/database-management-system/

- Madhav Lal Sukhadia University (MLSU). (n.d.). *Overview, Objectives and Types of ISAR System*. Retrieved from https://mlsu.ac.in/econtents/5701_DL-BLISc%20%20ISAR%20Mening.pdf
- IGNOU. (n.d.). *Block 4: Information Storage and Retrieval Systems*. Retrieved from https://egyankosh.ac.in/bitstream/123456789/76438/1/Block-4.pdf
- Sprinkle Data. (2024). *Database Models in DBMS: A Comprehensive Guide*. Retrieved from https://www.sprinkledata.com/blogs/database-models-in-dbms-a-comprehensive-guide
- Databricks. (2026). *ACID Transactions in Databases*. Retrieved from https://www.databricks.com/glossary/acid-transactions
- Scribd. (2026). *Database Language*. Retrieved from https://www.scribd.com/document/980310425/5-Database-Language

Further Readings

- Date, C. J. (2004). *An Introduction to Database Systems* (8th ed.). Pearson Education.
- Navathe, S. B., and Elmasri, R. (2015). *Fundamentals of Database Systems* (7th ed.). Pearson.
- Chowdhury, G. G. (2004). *Introduction to Modern Information Retrieval* (2nd ed.). Facet Publishing.
- Bhatt, R. K. (2017). *Database Management Systems*. Ess Ess Publications.
- Ramakrishnan, R., and Gehrke, J. (2003). *Database Management Systems* (3rd ed.). McGraw-Hill.

IGNOU. (n.d.). *Unit 1: Database: Concept and Components*. Retrieved from https://www.egyankosh.ac.in/bitstream/123456789/76902/1/Unit-1.pdf

*** ***** ***

UNIT 4: DATABASE MODELLING SYSTEM

UNIT STRUCTURE

4.1 Learning Objective
4.2 Introduction to Database Modelling
4.3 Linear Sequential Model
4.4 Hierarchical Model
4.5 Network Model
4.6 Entity-Relationship (E-R) Model
4.7 Relational Model
4.8 Object-Oriented Model
4.9 Let Us Sum Up
4.10 Answer to Check Your Progress
4.11 Model Questions
4.12 References and Further Readings

4.1 Learning Objectives

After going through this unit, you will be able to:

- Understand the concept and purpose of database modelling in the context of information retrieval systems.
- Distinguish between the different types of database models, including the linear sequential, hierarchical, network, entity-relationship, relational, and object-oriented models.
- Analyse the structural characteristics, advantages, and limitations of each model.
- Relate the appropriate use of each model to real-world information management scenarios.

4.2 INTRODUCTION TO DATABASE MODELLING

A database modelling system is foundational to every information retrieval (IR) environment because it defines how data is structured, stored, and made accessible to users and applications. Before individual records can be searched, sorted, or linked, a model must govern their logical organisation. Without a clearly defined data model, an IR system would lack the structural coherence necessary for consistent and efficient retrieval.

"According to Steve Hoberman in *Data Modeling Fundamentals*, 'data modeling is the process of learning about the data, and the data model is the end result of the

data modeling process'". A data model, therefore, is not merely a technical blueprint; it is a conceptual representation that captures the nature of entities, events, and their associations within an organisation. A database model, as a specific type of data model, determines the logical structure of a database and fundamentally shapes how data can be stored, organised, and manipulated.

Data exists in the form of facts, figures, and other discrete bits of knowledge. When a large volume of such data items must be organised for retrieval, they must be arranged according to a defined schema that mirrors real-world concepts and structures. Data modeling defines not only data elements but also their internal structures and the relationships among them. The choice of database model directly influences the performance, scalability, and usability of any information retrieval system, making an understanding of these models essential for information professionals.

Prior to the 1980s, hierarchical and network models dominated database design. Since then, the field has diversified considerably, and the following models are now widely recognised: the linear sequential model, hierarchical model, network model, entity-relationship (E-R) model, relational model, and object-oriented model. Each model reflects a different way of conceptualising data and suits different types of information retrieval applications.

4.3 LINEAR SEQUENTIAL MODEL

The linear sequential model, also referred to as the flat model or flat file model, is the simplest and most foundational form of data organisation. It consists of a single, two-dimensional array of data elements arranged in a list or table format, with no hierarchical or networked relationships among the entries. The structure is essentially linear: one record follows another in sequence, and each record is treated as independent of all others.

A common practical example of this model is a class list containing only a student number and a name. In such a list, there are no parent-child relationships, no cross-references, and no linked records. The model is suitable only where data is uniform and relationships between entities are absent or unnecessary. For example, a simple inventory of book titles in a small reading room, where each entry requires no more than a few fields such as accession number and title, could be managed using a flat file.

While the simplicity of the flat model makes it easy to implement and understand, it is unsuitable for complex information retrieval environments. It cannot accommodate relational data, struggles with redundancy, and does not support multi-entity queries. As IR systems grew in sophistication, the limitations of the flat model prompted the development of more structured approaches. Nevertheless, the flat model remains in use in specific, limited contexts where lightweight data storage is sufficient, such as configuration files, simple lookup tables, and basic transaction logs.

4.4 HIERARCHICAL MODEL

The hierarchical model organises data in a tree-like structure, where each record has a single parent and may have one or more children. The topmost node in the structure is referred to as the root, and all other nodes branch downward in a parent-to-child relationship. This model is particularly well-suited for data that has a natural parent-child or one-to-many relationship.

"According to Barve, Sunita, 'a hierarchical model consists of a collection of records which are connected to each other through links,' and the tree-type data structure is used to represent such hierarchical relationships".

Hierarchical structures were widely employed in early mainframe database management systems. A prominent example is the Information Management System (IMS) developed by IBM, which used hierarchical organisation to manage large volumes of enterprise data. The model also underpins the structure of XML documents, which use nested element hierarchies to encode data. It efficiently describes many real-world relationships such as recipes, tables of contents, and nested organisational charts.

However, the hierarchical model has a significant limitation: it supports only one-to-many relationships. It cannot accommodate many-to-many relationships without substantial data redundancy. For example, a student enrolled in multiple courses would require duplication of student records under each course node. This structural rigidity prompted the development of the network model, which extended the hierarchical approach to handle more complex data relationships.

4.5 NETWORK MODEL

The network model was developed specifically to address the limitations of the hierarchical model, particularly its inability to represent many-to-many relationships without redundancy. In the network model, relationships are represented in terms of sets rather than strict hierarchies, allowing a record to have multiple parents and multiple children, thereby forming a more complex graph-like structure.

"According to Barve, 'the network model organises data using two fundamental concepts, called records and sets. Records contain fields, and sets define one-to-many relationships between records: one owner, many members'". Crucially, a record may be an owner in any number of sets and simultaneously a member in any number of sets, giving the model its characteristic flexibility. This contrasts sharply with the hierarchical model, where a child record can have only one parent.

The network model was standardised by the Conference on Data Systems Languages (CODASYL) and was widely implemented during the 1970s. It offered a significant advance over the hierarchical model by reducing data redundancy and enabling more realistic representations of complex data relationships. For instance, in a library setting, a single author may be associated with multiple books, and a single

book may be attributed to multiple authors; this many-to-many relationship is naturally handled in the network model.

Despite these advantages, the network model has practical drawbacks. Its structure is complex, and navigating the data requires explicit traversal of set relationships. Any change to the data structure frequently requires extensive reconfiguration, making maintenance difficult. The lack of a straightforward query mechanism further limits its accessibility, particularly for non-technical users. These challenges contributed to the rise of the relational model, which offered a more intuitive and flexible approach to data management.

Check Your Progress 1

Q1. What is the simplest type of database model?

Q2. Name the data structure used to represent the hierarchical model.

Q3. Match the following:

Column A	Column B
(a) Flat model	(i) Many-to-many relationships
(b) Hierarchical model	(ii) Two-dimensional array
(c) Network model	(iii) One-to-many, tree structure

Q4. Fill in the blank: The topmost node in a hierarchical model is called the _____.

Q5. The network model was standardised by which organisation?

4.6 ENTITY-RELATIONSHIP (E-R) MODEL

The Entity-Relationship model is a conceptual data model that views the real world in terms of entities, their attributes, and the relationships among them. It was originally developed by Peter Pin-Shan Chen and published in his landmark 1976 paper titled "The Entity-Relationship Model: Toward a Unified View of Data," published in *ACM Transactions on Database Systems*. The E-R model was designed to unify the network and relational views of data and to provide a natural, intuitive means of representing the logical structure of a database.acm+1

"According to Peter Chen in his 1976 paper in *ACM Transactions on Database Systems*, 'the entity-relationship model adopts the more natural view that the real world consists of entities and relationships. It incorporates some of the important semantic information about the real world'". An entity is any distinguishable object or concept in the domain being modelled, such as a book, an author, a publisher, or a library user. An attribute is a property of an entity (for example, the title of a book or the birth year of an author), and a relationship describes the association between two or more entities (for example, "authored by" connecting a book to an author).

E-R diagrams are the primary tool used to visually represent these data concepts. The diagrams use standard notation: rectangles denote entities, ovals denote attributes, and diamonds denote relationships. This graphical representation

makes it easier for database designers, stakeholders, and users without technical backgrounds to understand and validate the data structure before implementation. The clarity and expressiveness of E-R diagrams have made them a standard instrument in database design practice.

The E-R model supports the representation of strong entities, weak entities, single-valued and multi-valued attributes, as well as generalisation and specialisation hierarchies. These constructs allow a designer to capture nuanced real-world semantics. For instance, in an information retrieval context, an E-R model for a bibliographic database might represent entities such as Document, Author, Subject Heading, and Publisher, along with their attributes and inter-relationships. The E-R model does not specify a physical storage mechanism, it functions at the conceptual level, serving as a bridge between high-level user requirements and the logical schemas of implementable models such as the relational model.

The E-R model's influence extends beyond traditional relational databases. It has shaped the development of software engineering tools, including Computer-Aided Software Engineering (CASE) tools and the Unified Modeling Language (UML), which adapted E-R principles into object-oriented design notation. Today, the E-R model is widely used in database design instruction and remains a foundational framework in the information sciences.

4.7 RELATIONAL MODEL

The relational model is the most widely adopted database model in contemporary information management. It was proposed by Dr. Edgar Frank Codd in 1970 in his seminal paper "A Relational Model of Data for Large Shared Data Banks," published in *Communications of the ACM*. Codd's objective was to make database management systems more independent of any particular application by providing a simple, mathematically grounded framework for data representation.

"According to Barve (2016), 'the relational model was introduced in 1970 as a way to make database management systems more independent of any particular application. Three key terms are used extensively in relational database models: relations, attributes, and domains'". A relation is a table composed of rows and columns. Each column represents an attribute, and the domain defines the permissible values that an attribute may take. Each row, referred to as a tuple, represents a single instance of the entity described by the relation.

The basic data structure of the relational model is the table. For example, a relation called EMPLOYEE would have columns for attributes such as Name, Employee ID, and Department, and each row would represent a specific employee. A relational database typically contains multiple such tables, and the power of the model lies in the ability to establish logical connections between tables through shared attribute values, known as keys. A primary key uniquely identifies each tuple within a relation, while a foreign key establishes a referential link to a primary key in another relation.

The relational model enforces two fundamental integrity constraints: entity integrity, which requires that primary keys be non-null, and referential integrity, which requires that a foreign key value in one relation must correspond to an existing primary key value in another. These constraints ensure consistency and reliability across the database, which is especially important in large-scale information retrieval environments where cross-referencing between records is frequent.

The query language most commonly associated with the relational model is Structured Query Language (SQL), which allows users to retrieve, insert, update, and delete data using a declarative syntax. SQL's accessibility has made the relational model the dominant choice for bibliographic databases, library management systems, and enterprise information systems alike. The model's mathematical foundation in set theory and first-order predicate logic also lends it considerable theoretical rigour. Despite the emergence of newer models, the relational model continues to serve as the backbone of most production-grade IR systems.

4.8 OBJECT-ORIENTED MODEL

The object-oriented data model (OODM) represents a significant departure from the table-based architecture of the relational model. It was developed to address the limitations of relational systems when dealing with complex, non-uniform data types such as images, audio, video, and other multimedia content that are increasingly common in modern IR environments. The OODM is considered a semantic data model because it directly encodes the meaning and behaviour of data within the object structure.

"According to Barve (2016), 'object-oriented data model (OODM) is the basis for the object-oriented database management system (OODBMS). Object includes information about relationships between facts within the object, and relationships with other objects'". In the OODM, data is represented as objects, which are self-contained units that bundle together both data (attributes) and the operations that can be performed on that data (methods). This encapsulation means that each object carries its own operational logic, unlike the relational model where data and processing are separated.

The key features of the object-oriented data model include:

- **Object identity:** Each object is assigned a unique, system-generated object identifier that distinguishes it from all other objects in the database.
- **Encapsulation:** Data and methods are packaged together within the object, hiding internal implementation details from outside users.
- **Abstraction:** Only the essential aspects of an entity are modelled; irrelevant properties are excluded.
- **Inheritance:** Objects can inherit attributes and methods from parent classes, enabling reuse and reducing redundancy in class definitions.
- **Classes:** A class is a template that defines the structure and behaviour shared by a group of similar objects.

- **Relationships:** Complex relationships such as aggregation, generalisation, and specialisation can be modelled explicitly within the OODM framework.

The OODM provides four primary types of data operations: defining schemas, creating databases, retrieving objects, and expanding objects. An object-oriented database management system (OODBMS) that implements the OODM offers several advantages over conventional relational DBMSs, including richer semantic expression, support for complex objects such as multimedia documents, better performance in complex retrieval applications, and greater extensibility of data types.

The OODM is particularly relevant for digital library systems that must manage diverse media types. For example, a digital archive of research outputs might include textual documents, datasets, video recordings of lectures, and image collections. A relational model would struggle to represent these efficiently, whereas an OODM could encode each resource type as a distinct class with type-specific attributes and retrieval methods. The OODM thus aligns closely with the demands of contemporary, heterogeneous IR environments.However, the OODM is more complex to design and implement than the relational model, and the absence of a universally adopted query language comparable to SQL has limited its mainstream adoption. Most modern systems instead employ object-relational databases, which extend the relational model with object-oriented constructs, achieving a practical balance between the two approaches.

Check Your Progress 2

Q1. Who proposed the Entity-Relationship model and in which year?

Q2. Fill in the blank: In the relational model, a row in a table is called a .

Q3. Match the following:

Column A	**Column B**
(a) E-R Model	(i) E.F. Codd, 1970
(b) Relational Model	(ii) Tables, tuples, attributes
(c) Object-Oriented Model	(iii) Peter Chen, 1976
(d) Relational Model	(iv) Encapsulation, inheritance

Q4. What does SQL stand for?

Q5. Name any two key features of the object-oriented data model.

Q6. True or False: The E-R model specifies how data is physically stored in a database.

4.9 LET US SUM UP

We have learned the concept of database modelling as a foundational element of Information Retrieval (IR) systems. A database model defines the logical structure through which data is stored, organized, and retrieved. The unit examines six major database models in sequence. The linear sequential model is the simplest, storing data as a flat, two-dimensional list. The hierarchical model arranges records in a tree structure supporting one-to-many relationships. The network model extends this by allowing many-to-many relationships through set-based organization. The Entity-Relationship (E-R) model, proposed by Peter Chen in 1976, offers a conceptual framework using entities, attributes, and relationships represented through diagrams. The relational model, introduced by E.F. Codd in 1970, organises data into tables and uses SQL for querying. Finally, the object-oriented model encapsulates data and methods within objects, making it suitable for complex and multimedia-rich IR environments. Each model represents a progressive response to the limitations of its predecessor.

4.10 ANSWER TOCHECK YOUR PROGRESS

Check Your Progress 1

Ans to Q1. The linear sequential model (flat model) is the simplest type of database model.

Ans to Q2. The tree-type data structure is used to represent the hierarchical model.

Ans to Q3. (a) – (ii); (b) – (iii); (c) – (i)

Ans to Q4. Root

Ans to Q5. The network model was standardised by the Conference on Data Systems Languages (CODASYL).

Check Your Progress 2

Ans to Q1. The Entity-Relationship model was proposed by Peter Pin-Shan Chen in 1976.

Ans to Q2. Tuple

Ans to Q3. (a) – (iii); (b) – (i) and (ii); (c) – (iv)

Ans to Q4. SQL stands for Structured Query Language.

Ans to Q5. Any two of the following: object identity, encapsulation, abstraction, inheritance, classes.

Ans to Q6. False. The E-R model operates at the conceptual level and does not specify physical storage mechanisms.

4.11 MODEL QUESTIONS

Short Questions

1. Define a data model and state its significance in an information retrieval system.
2. What is the flat file model? Give one example of its application.
3. Distinguish between the hierarchical model and the network model.
4. What is an E-R diagram? Name its three primary components.
5. State any two integrity constraints enforced in the relational model.
6. What is encapsulation in the context of the object-oriented data model?
7. Name the query language most commonly used with the relational model.
8. Who developed the hierarchical IMS system and for which organisation?
9. Define a "tuple" as used in the relational model.

Long Questions

1. Describe the hierarchical database model in detail. Discuss its structure, advantages, and limitations with suitable examples from information retrieval environments.
2. Explain the Entity-Relationship model.
3. Compare and contrast the relational model and the object-oriented data model. In what types of IR environments is each model most appropriately applied?
4. What is data modeling? Explain the role of database modelling systems in the design and functioning of Information Storage and Retrieval (ISAR) systems.

4.12 REFERENCES AND FURTHER READINGS

References

- Barve, Sunita. (2016). *ISAR Systems: Functions and Design*. Module 6. E-PG Pathshala, INFLIBNET Centre.
- Chen, Peter P. S. (1976). The Entity-Relationship Model: Toward a Unified View of Data. *ACM Transactions on Database Systems*, 1(1), 9–36.
- Codd, Edgar F. (1970). A Relational Model of Data for Large Shared Data Banks. *Communications of the ACM*, 13(6), 377–387.
- Date, C. J. (2003). *An Introduction to Database Systems* (8th ed.). Addison-Wesley.
- Hoberman, Steve. *Data Modeling Fundamentals*. IRM UK. Available at: www.irmuk.co.uk/events/72.cfm
- Elmasri, Ramez, and Navathe, Shamkant B. (2015). *Fundamentals of Database Systems* (7th ed.). Pearson.

- OWOX BI. (2025). *Object-Oriented Data Models: Guide and Examples*.
- Silberschatz, Abraham, Korth, Henry F., and Sudarshan, S. (2019). *Database System Concepts* (7th ed.). McGraw-Hill.

🔍 Further Readings

- Elmasri, R., & Navathe, S. B. (2015). *Fundamentals of database systems* (7th ed.). Pearson Education.
- Chen, P. P. S. (1976). The entity-relationship model: Toward a unified view of data. *ACM Transactions on Database Systems*, *1*(1), 9–36. https://doi.org/10.1145/320434.320440
- Silberschatz, A., Korth, H. F., & Sudarshan, S. (2019). *Database system concepts* (7th ed.). McGraw-Hill Education.

*** ***** ***

UNIT 5: ORGANISATION OF DATA

UNIT STRUCTURE

5.1 Learning Objectives
5.2 Introduction
5.3 Data Organisation
5.4 Record Structure
5.5 File Structure
    5.5.1 Order of Records
    5.5.2 Finding a Record
5.6 Organisational Methods
    5.6.1 Sequential File Structure
    5.6.2 Direct File Organisation
    5.6.3 Index File Structure
    5.6.4 Tree Structures and B-Trees
5.7 Let Us Sum Up
5.8 Answer to Check Your Progress
5.9 Model Questions
5.10 References and Further Readings

5.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- Explain the concept of data organisation and its significance in information retrieval systems.
- Describe the structure of a record and a file in the context of database management.
- Distinguish between sequential and direct file organisation, identifying the advantages and limitations of each.
- Explain how index file structures and tree-based structures (including B-trees) facilitate efficient data retrieval.
- Apply knowledge of file organisational methods to evaluate their suitability for different retrieval contexts.

5.2 INTRODUCTION

The physical organisation of data is one of the foundational concerns in information retrieval. How records and files are arranged on storage media determines,

to a significant degree, how efficiently those records can be located and retrieved. As information systems have grown in scale and complexity, the methods used to organise data have evolved from simple sequential arrangements to sophisticated index and tree structures. Understanding these methods is essential for designing systems that can retrieve information with minimal delay, even from large and complex databases.

According to Silberschatz, Korth, and Sudarshan in *Database System Concepts* (1997), the physical organisation of data directly affects the time required to access, write, or modify a record. The choice of organisation method is therefore not merely a technical decision but also a functional one, shaped by the types of operations a system is expected to perform and the volume of data it must handle. This unit examines the principal methods of data organisation, with particular attention to their structures, operating principles, and comparative strengths and weaknesses within the context of information retrieval.

5.3 DATA ORGANISATION

Data organisation refers to the manner in which data is structured and stored in a computer file system or database so that it can be efficiently accessed, managed, and retrieved. Every information retrieval system depends on an underlying structure that governs how data elements are stored, identified, and located. This structure affects the time complexity and resource requirements of retrieval operations.

A file, in the context of data management, is a collection of related records stored as an individual entity on a storage medium. According to Aditya Tripathi in *Physical Organisation of Data*, "A file is a collection of bytes stored as an individual entity... database files are made up of a series of records." The data within a file pertains to entities, where an entity is anything about which information can be stored, such as a person, object, event, or concept. Each entity has attributes, whose values characterise particular instances. For example, a book stored in a library database may have attributes such as title, author, publisher, and price.

Each attribute of an entity is represented in storage by a data item. A set of data items that collectively describe one instance of an entity is called a record. A collection of related records constitutes a file. This hierarchical model, from data item to record to file, forms the basic organisational unit of all database and information retrieval systems.

5.4 RECORD STRUCTURE

A record is the elementary unit of stored data in a file. It is composed of fields, each field corresponding to one attribute of the entity being described. Record structure refers to the manner in which these fields are arranged and delimited within the record.

According to Tripathi, "It is the structure where all the elements of a record are arranged in an organized manner to provide a structure to data. Record is the simplest form to store all instances of attributes of elements whose meaningful organized form

gives a meaningful data." A record structure typically consists of fixed-length records, a fixed number of fields, and a length indicator. It may use an index to keep track of the address of each record and place a delimiter at the end to mark its boundary.

Records may be of fixed length or variable length. In fixed-length records, every field occupies the same amount of storage space in every record. This simplifies addressing and retrieval but may waste storage space when fields do not use their full allotted length. Variable-length records use only as much space as is required, improving storage efficiency, but require additional mechanisms such as delimiters or length fields to parse them correctly. The choice between fixed and variable record structures depends on the nature of the data and the access patterns of the system.

A key concept in record structure is the **primary key**, which is a field or set of fields whose value is unique to a single record, thereby serving as the identifier for that record. As noted by Tripathi, "A primary key is a field or set of fields whose content is unique to one record and can therefore be used to identify that record. Primary keys usually include student ID, title of book, etc." The primary key plays a central role in retrieval operations, as it provides the mechanism by which the system locates and distinguishes individual records.

5.5 FILE STRUCTURE

A file structure describes how records are organised, stored, and accessed within a file. "File structure basically deals with the order and arrangement of files by computers within a database." Different file structures support different access patterns, and the choice of structure has significant implications for retrieval efficiency.

5.5.1 Order of Records

The order of records refers to the sequence in which records are stored in a file. Records may be stored in the order in which they were received (insertion order), or they may be arranged according to the values of one or more key fields. According to Tripathi, "An ordered record is helpful to access data easily from a database. For example, information of employees of a library is arranged according to their job positions." Ordering by key value supports efficient sequential retrieval and simplifies range-based searches.

5.5.2 Finding a Record

Record finding, or retrieval, is the process of locating and returning a specific record from a file. It usually involves specifying a key value and then applying a search strategy to find the matching record. Tripathi states that "Usually a key field is assigned and the record may be retrieved by that key. A record may contain information about a book like title, author, ISBN, publication and title may be assigned as a key." The

efficiency of record finding depends on the file organisation method in use, the size of the file, and whether an index or other access structure is available.

5.6 ORGANISATIONAL METHODS

The method of file organisation chosen when data is stored on a storage device determines how the data can later be accessed. "The organization of data in a file is influenced by a number of factors, but the most important factor among them is the time required to access a record, to write a record or to modify a record." Each file organisation method offers advantages in some types of operations and limitations in others. The three principal methods considered in this unit are sequential file organisation, direct file organisation, and index-based organisation.

5.6.1 Sequential File Structure

Sequential file organisation is a method in which records are stored one after another in a predetermined order, typically determined by the value of the primary key. All records, which may be of some uniform length, are arranged consecutively in physical storage. "In sequential file organization, records are stored in some predetermined sequence, one after the other. One field, referred to as the primary key, usually determines their sequence or order."

In a sequential file, locating a specific record requires starting at the beginning of the file and reading through records one by one, comparing each key to the search key. There is no need to know the precise storage address of any record; the system proceeds in order until the desired record is found or the end of the file is reached. This linear scan characteristic gives sequential files their defining property: simplicity of access logic combined with predictable traversal behaviour.

Advantages of Sequential File Organisation

The following are the main advantages of sequential file organisation:

- It is the most efficient form of organisation when the entire file, or most of it, must be processed at once, as in batch processing.
- The transaction file and master file together act as a backup, and can be used to create a new master file if the original is damaged or corrupted.
- The sequential structure is simple to implement and requires no additional overhead for index management.
- It performs well on storage devices where sequential access is the native mode of operation.

Disadvantages of Sequential File Organisation

The following aredisadvantages:

- It is time-consuming to access a particular record because the system may be required to read through many records before reaching the target.
- The entire file must be processed and a new master file created even if only one record requires updating, resulting in significant overhead.
- Accessing a single record located in the middle of a large file requires traversal of all preceding records, making it inefficient for random access operations.
- It is not suitable for applications that require frequent random access or real-time updating of individual records.

Sequential file organisation remains relevant in applications such as payroll processing, billing, and audit logging, where batch processing of the entire dataset is routine and real-time access is not required.

5.6.2 Direct File Organisation

Direct file organisation, also referred to as random file organisation, is designed to support random access and rapid, non-sequential retrieval of individual records. According to Tripathi, "Direct file organization is designed to provide random access and rapid direct non-sequential access to records. Using this organization, records are inserted in what appears to be a random order, not in sequence by key field value."

In this method, each record is assigned a relative storage address derived from the value of a key field. When a record is to be stored, a mathematical function (commonly a hash function) is applied to the key value to compute a target storage address, and the record is placed at that address. During retrieval, the same function is applied to the supplied key to calculate where the record should be located, and the system retrieves it directly from that address, without scanning preceding records.

Direct file organisation is especially well suited to interactive systems where individual records must be updated or retrieved promptly. "In direct processing system, data are directly input into the system through a terminal. The system locates the specific record in master file and then updates it. Direct access systems do not search the entire file rather, they move directly to the needed record."

Advantages of Direct File Organisation

The advantages include:

- Data can be accessed directly and quickly without sequential scanning.
- Primary and secondary indexes can be used to support multiple access paths.
- Centrally maintained data can be updated easily without recreating the entire file.
- It is well suited to real-time applications requiring rapid individual record access.

Disadvantages of Direct File Organisation

The disadvantages identified are:

- The use of an index lowers overall computer system efficiency due to additional processing overhead.
- Because files are updated directly, there may be no backup if a file is destroyed.
- Regular creation of backup files is required to protect data integrity.
- Address collision (where two keys map to the same address) requires collision resolution strategies, adding complexity to implementation.

5.6.3 Index File Structure

An index file structure is an auxiliary mechanism that improves the efficiency of record searches within a data file. An index in this context is a separate, smaller file that maintains a mapping between key values and the storage addresses of corresponding records. According to Kowalski in *Information Storage and Retrieval Systems: Theory and Implementation*, index files are a central component of information retrieval systems, noting that "A good analogy to an index file is the card catalogue in a library."

Tripathi defines the role of an index in data access: "An index file is an auxiliary file that makes it more efficient to search for a record in the data file. An index is also known as access path on the field which is usually specified. Index file is stored on disk and occupies less space than data file due to its much smaller entries."

Because the index stores only the key and address rather than the full record, it is considerably smaller than the data file and can be searched more rapidly. Once the index is consulted and the target address is identified, the system retrieves the record directly from the data file at that address. This two-step process (index lookup followed by direct record access) is substantially faster than sequential scanning for most access patterns, particularly in large files.

Indexed file structures support multiple access paths when secondary indexes are created on non-primary fields. For example, a library employee database may be indexed by Employee ID, Name, Section, or Designation, allowing users to search by any of these attributes. This flexibility distinguishes indexed organisation from purely sequential or purely direct organisation.

The inverted file structure is a particularly important application of indexing in information retrieval. As Kowalski explains, "Inversion list file structures are well suited to store concepts and their relationships. Each inversion list can be thought of as representing a particular concept. The inversion list is then a concordance of all of the items that contain that concept." Inverted files minimise secondary storage access when multiple search terms are applied across a large database, and are used in virtually all commercial information retrieval systems.

5.6.4 Tree Structures and B-Trees

Tree structures provide a dynamic and hierarchical approach to organising index data, enabling efficient search, insertion, and deletion operations. According to Tripathi, "Tree is a structure which consists of nodes or vertices containing node information together with pointers giving access to additional nodes of the tree. A tree organization supports operations such as searching for a record, inserting a new record and deleting a record. A tree search is performed by comparing the search key with the key values attached to certain nodes of the tree, starting with the root of the tree."

The basic tree types in data organisation include binary search trees and balanced trees. In a binary search tree, each node has at most two children, and the key values are ordered such that the left subtree contains smaller values and the right subtree contains larger values. While binary search trees are effective for relatively small datasets, their performance degrades when the tree becomes unbalanced, causing search paths to lengthen unpredictably.

B-Trees

The B-tree (Balanced tree) addresses this limitation by generalising the binary search tree to allow nodes to have more than two children. "A B-tree is a data structure that provides sorted data and allows searches, sequential access, attachments and removals in sorted order. The B-tree is highly capable of storing systems that write large blocks of data. The B-tree simplifies the binary search tree by allowing nodes with more than two children." In a B-tree of order *m*, each internal node may contain up to *m-1* keys and *m* child pointers, and the tree remains balanced at all times through automatic splitting and merging of nodes during insertions and deletions.

The B-tree achieves its efficiency through two key properties: balance and branching factor. Balance ensures that all leaf nodes are at the same depth, so the maximum number of comparisons required to locate any record is equal to the height of the tree. A high branching factor (many keys per node) keeps the tree shallow, reducing the number of disk accesses needed for retrieval. As PlanetScale's database reference notes, "When searching in this way, you only need to visit one node at each level of the tree to search for one key. Therefore, the fewer levels it has (or the shallower it is), the faster searching can be performed." Search time complexity in a B-tree is O(log n), making it highly efficient even for very large datasets.

B+ Trees

A commonly used variant is the B+ tree, in which all data records are stored only in the leaf nodes, while internal nodes store only keys used for routing searches. "All records are stored in the leaf nodes of the B+ tree. No records are stored on non-leaf nodes. Each of the leaf nodes references the next record in the tree." This design means that sequential scans through the data can be performed efficiently by traversing the linked list of leaf nodes, without visiting internal nodes. The B+ tree is the preferred index structure in most modern database management systems due to its

support for both point queries (finding a specific key) and range queries (finding all keys within a specified interval).

B-trees and B+ trees are used extensively in information retrieval to maintain inverted file indexes, as the ordered structure supports fast lookup of index terms and their associated inversion lists. Their self-balancing property ensures that performance does not degrade as data is inserted or deleted over time.

Check Your Progress – I

Q1. What is a primary key in a file record?

Q2. Match the following:

Column A	Column B
(a) Sequential file	(i) Hash function used for address calculation
(b) Direct file	(ii) Auxiliary file mapping keys to addresses
(c) Index file	(iii) Records stored one after another by key order

Q3. Fill in the blank: In sequential file organisation, locating any record is performed by starting at the of the file and reading each record in sequence.

Q4. Which file organisation method is most suitable for batch processing applications?

Q5. True or False: An index file typically occupies more storage space than its corresponding data file.

Q6. Define a B-tree in one sentence.

Q7. Fill in the blank: In a B+ tree, all data records are stored only at the nodes.

Q8. What does the search time complexity O(log n) indicate about B-tree performance?

5.7 LET US SUM UP

Data organisation encompasses the methods by which records and files are arranged, stored, and accessed in a database or file system. A record is the basic unit of stored information, composed of fields that correspond to the attributes of an entity. A collection of records constitutes a file, and the structure of that file determines the efficiency of retrieval operations.

Sequential file organisation stores records consecutively in key order and is well suited to batch processing, but it is inefficient for random access. Direct file organisation uses mathematical functions to map key values to storage addresses, enabling rapid random access but requiring careful management of collisions and backups. Index file structures maintain a separate, compact mapping of keys to record addresses, substantially reducing retrieval time and supporting multiple access paths.

Tree structures, particularly B-trees and B+ trees, provide dynamic, self-balancing index mechanisms that guarantee logarithmic search time regardless of the file size, making them the preferred choice in most large-scale information retrieval and database systems.

The organisational method chosen for any given system must reflect the operational requirements of that system: the volume of data, the frequency and type of queries, the acceptable retrieval time, and the storage resources available. No single method is universally optimal; each offers a different balance of advantages and trade-offs suited to different retrieval contexts.

✓ 5.8 ANSWER TO CHECK YOUR PROGRESS

Ans to Q1: A primary key is a field or set of fields whose value is unique to one record and can therefore be used to identify that record. Examples include student ID or ISBN.

Ans to Q2: (a) – (iii); (b) – (i); (c) – (ii)

Ans to Q3: Beginning

Ans to Q4: Sequential file organisation is most suitable for batch processing, as it is most efficient when the entire file or most of it must be processed at once.

Ans to Q5: False. An index file occupies less space than the data file because it stores only key values and addresses, not full records.

Ans to Q6: A B-tree is a self-balancing tree data structure that maintains sorted data and allows efficient searches, sequential access, insertions, and deletions.

Ans to Q7: Leaf

Ans to Q8: O(log n) indicates that the number of comparisons needed to retrieve a record grows logarithmically with the size of the dataset, meaning retrieval remains fast even as the database grows very large.

? 5.9 MODEL QUESTIONS

Short Questions

1. Define the term "record structure" as used in database management.
2. What is a primary key, and why is it important in information retrieval?
3. State two advantages of sequential file organisation.
4. What is a direct file organisation? How is a record's storage address determined in this method?
5. Define an index file. How does it differ from the data file it references?
6. What is a B-tree? State its order and explain the significance of balance in this structure.

7. What is the main structural difference between a B-tree and a B+ tree?
8. State the time complexity of search in a B-tree and explain its significance.
9. What is an inverted file structure and what is its role in information retrieval?

Long Questions

1. Explain the concept of sequential file organisation in detail. Discuss its advantages and disadvantages.
2. Describe direct file organisation with reference to how records are stored and retrieved.
3. What is an index file structure? Explain how index files improve retrieval efficiency and describe the role of primary and secondary indexes.

5.10 REFERENCES AND FURTHER READINGS

References

- Tripathi, Aditya. *Physical Organisation of Data* (Module 4). New Delhi: Indira Gandhi National Open University.
- Kowalski, G. J. (2002). *Information Storage and Retrieval Systems: Theory and Implementation* (2nd ed.). Boston: Kluwer Academic Publishers.Information-Storage-And-Retrieval-Systems-Theory-And-Impl-2e_Kowalski-GJ-2002.pdf
- Silberschatz, A., Korth, H. F., and Sudarshan, S. (1997). *Database System Concepts*. New York: McGraw-Hill.
- Ramakrishnan, R. and Gehrke, J. *Database Management Systems* (2nd ed.). University of Wisconsin, Madison: McGraw-Hill.
- Robbins, Robert J. (1994). *Database Fundamentals*. Baltimore: Johns Hopkins University.
- Healey, R. G. *Database Management Systems*. Leicester: University of Leicester.
- Haithcoat, Tim. *Relational Database Management Systems: Database Design and GIS*. Columbia: University of Missouri.

Further Readings

- Knuth, D. E. (1973). *The Art of Computer Programming, Volume 3: Sorting and Searching*. Reading, MA: Addison-Wesley. (Provides foundational treatment of B-trees and balanced search trees.)
- Frakes, W. B. and Baeza-Yates, R. (Eds.) (1992). *Information Retrieval: Data Structures and Algorithms*. Englewood Cliffs, NJ: Prentice Hall.
- Witten, I. H., Moffat, A., and Bell, T. C. (1999). *Managing Gigabytes: Compressing and Indexing Documents and Images* (2nd ed.). San Francisco:

Morgan Kaufmann. (Covers inverted file structures and index compression in depth.)

- Built In. (2023). "How Database B-Tree Indexing Works." https://builtin.com/data-science/b-tree-index
- Baeldung. (2024). "File Access: Sequential vs. Direct vs. Indexed." Available at: https://www.baeldung.com/cs/file-access
- PlanetScale. (2024). "B-trees and Database Indexes." https://planetscale.com/blog/btrees-and-database-indexes

*** ***** ***

UNIT 6: INDEXING IN INFORMATION RETRIEVAL

UNIT STRUCTURE

6.1 Learning Objectives
6.2 Introduction
6.3 Document Representations
6.4 Query Formulation
6.5 Query Matching
6.6 Relational Statements in Query Specification
6.7 Relevance Feedback
6.8 Let Us Sum Up
6.9 Answer to Check Your Progress
6.10 Model Questions
6.11 References and Further Readings

6.1 Learning Objectives

After going through this unit, you will be able to:

- Explain the concept of document representation and identify its major types and weighting mechanisms.
- Describe the process of query formulation in information retrieval systems.
- Distinguish between exact match and ranked retrieval approaches to query matching.
- Understand the role of relational operators and Boolean logic in query specification.
- Explain the concept and methods of relevance feedback in iterative retrieval.

6.2 INTRODUCTION

Indexing is the foundational process that enables an information retrieval (IR) system to respond meaningfully to user queries. At its core, an IR system must perform three interdependent operations: it must represent the content of documents in a form that the system can process; it must translate a user's information need into a corresponding formal query; and it must match document representations against query representations to identify relevant items. These three operations are tightly intertwined, and the quality of each determines the overall effectiveness of the retrieval system.

"According to Chowdhury in *Introduction to Modern Information Retrieval*, the different functions of information retrieval systems include analysing the contents of sources, representing those contents in a way that matches user queries, analysing user queries and representing them in a form suitable for matching the database, and matching the search statement with the stored database."

The present unit examines each of these core components in sequence, beginning with how documents are structurally represented for indexing, moving through the formulation of queries, the mechanisms by which queries are matched against document representations, the specification of relational statements, and finally, the process by which user feedback is incorporated to refine and improve retrieval results.

6.3DOCUMENT REPRESENTATIONS

6.3.1 The Nature of Document Representation

Document representation is the process by which the intellectual content of a document is converted into a form that an IR system can search and compare. "According to Panigrahi in *Basic Concepts and Components of Information Retrieval Systems* (2016), indexing, also referred to as cataloguing or metadata assignment from the library science point of view, is the manual or automated process of creating indexes for record collections, enabling researchers to more quickly find records without having to look through hundreds or thousands of records."

Subject indexing, in particular, is concerned with identifying and describing the subject content of documents. It operates at three distinct levels: terms within a document, objects within a collection, and documents within a broad field of knowledge. The objective is to produce an index that accurately predicts the document's relevance for a given user query.

6.3.2 Manual and Automatic Indexing

Indexing may be performed either manually by trained indexers or automatically by algorithms. Manual indexing draws on human expertise to identify concepts that may not be explicitly stated in the text, but it is subject to inconsistency between different indexers. "According to Kowalski in *Information Storage and Retrieval Systems: Theory and Implementation* (2002), in an experiment on consistency in TREC-2, there was, on the average, a 20 per cent difference in judgment of the same items between the original and a second independent judge of over 400 items, indicating that since judgments on relevance differ, the selection of index terms and their weighting also differ."

Automatic indexing begins with the extraction of all words from a text, followed by the elimination of stop words, stemming, term counting, and optionally mapping terms to concepts using a thesaurus or ontology. Stop words are high-frequency function words such as "the," "of," and "and" that carry little discriminating information.

Stemming reduces inflected or derived word forms to their common root form, such that "computing," "computed," and "computation" would all reduce to the stem "comput."

6.3.3 Weighted and Unweighted Representations

Document representations in IR systems may be either unweighted or weighted. In an unweighted or binary indexing system, the representation records only the presence or absence of a term, treating all index terms as equally significant. "According to Kowalski (2002), in an unweighted indexing system, it is not possible to tell the difference between the main topics in the item and a casual reference to a concept."

In a weighted indexing system, each index term is assigned a numerical value that represents its relative importance in expressing the content of the document. The most widely used weighting formula is Term Frequency-Inverse Document Frequency (TF-IDF). Term Frequency (TF) captures the frequency of a term within a given document, operating on the premise, first proposed by Luhn, that the significance of a concept in a document is directly proportional to the frequency of its associated word. Inverse Document Frequency (IDF) adjusts this weight by considering how commonly a term appears across the entire document collection. The combined TF-IDF formula assigns high weights to terms that are frequent in a given document but rare across the collection as a whole, thereby producing index representations that are both specific and discriminating.

The vector space model, developed by Salton through the SMART system at Cornell University, formalizes this approach. Each document is represented as a vector in an n-dimensional space, where each dimension corresponds to a unique term in the collection vocabulary and the value at each position is the term's TF-IDF weight. A value of zero indicates the term's absence from the document.

6.3.4 Concept-Based Representations

Beyond term-based approaches, concept indexing aims to represent the latent semantic content of documents rather than merely their surface vocabulary. Latent Semantic Indexing (LSI) applies singular value decomposition to the term-document matrix, reducing the high-dimensional space of individual terms to a smaller number of orthogonal concept dimensions that capture underlying semantic associations. This allows the system to identify documents as relevant even when they do not share exact vocabulary with the query.

Check Your Progress 1

Q1. What is the primary purpose of subject indexing in an information retrieval system?

Q2. Fill in the blank: The process for reducing inflected words to their stem, base, or root form is called_____.

Q3. Name the weighting algorithm that considers both the frequency of a term in a document and its rarity across the document collection.

6.4 QUERY FORMULATION

6.4.1 The Query and Information Need

Query formulation is the process by which a user's information need is translated into a formal representation that the IR system can process. "According to Panigrahi (2016), information retrieval means making use of available information in order to anticipate the extent to which a given document is significant or useful for a particular user's information need as outlined in a free-form query description, also called a topic description or query statement."

A user's information need and the formal query that represents it are closely related but rarely identical. A user may have a vague or partially articulated need that they must first clarify before expressing it as a structured query. Throughout the search process, users typically refine and revise their queries as they review retrieved documents and develop a clearer understanding of what they are seeking.

6.4.2 Types of Query Language

Queries in IR systems are generally expressed in one of two broad forms of query language: procedural (command language) and non-procedural (natural language). "According to Barve in *Querying of the Information Retrieval System* (2016), a procedural language uses commands such that little or no knowledge is required for the IRS to find what was asked for and retrieve the same, whereas natural language queries generally tend to be ambiguous in syntax and meaning."

Command language queries are structured and unambiguous; they specify search terms and logical operators in a precise syntax that the system interprets directly. Natural language queries, by contrast, require additional processing to resolve ambiguity, including part-of-speech tagging, disambiguation, and semantic analysis.

6.4.3 Query Construction and Expansion

The construction of a useful query involves deciding which terms to include, how to combine them, and whether to expand them using synonyms or thesaurus relationships. An IR system may present a subject hierarchy for browsing to help users identify appropriate descriptors, or it may interactively ask clarifying questions and construct the query from the answers. The system can also suggest synonyms, narrower terms, and broader terms from its thesaurus, assisting the user in visualizing all relevant features of the topic.

Automatic query expansion uses the lexical and semantic structure of the index to extend a user's query with related terms. If a query asks for "dessert," the system may expand the search to find documents containing "sweets" as well. Hierarchic expansion allows inclusive searching so that a query for "fruit" may also retrieve documents about specific fruits even if the precise term "fruit" is absent.

A query may be structured in several ways: as a free-text search using unstructured term lists, as a Boolean query combining terms with logical operators, or as a weighted vector query in which terms are assigned importance values.

6.5 QUERY MATCHING

6.5.1 Principles of Matching

Query matching is the process by which the IR system predicts the relevance of documents to a given query by comparing their internal representations. The essential premise is that "the document relevance is predicted by identifying the relevant features of the query with that of the document". Matching approaches can be broadly classified as exact match or inexact (ranked) match, a distinction that reflects the degree of precision with which documents must satisfy query conditions.

"According to Kowalski (2002), Belkin and Croft (1987) categorize IR conceptual models by dividing retrieval techniques into exact match and inexact match; the exact match category contains text pattern search and Boolean search techniques, while the inexact match category contains such techniques as probabilistic, vector space, and clustering models."

6.5.2 Exact Match Retrieval

In an exact match system, documents are retrieved only if they completely satisfy all the conditions specified in the query. The most common form is the Boolean IR system, in which documents are represented by sets of keywords stored in an inverted file, and queries are composed of keywords connected by Boolean logical operators AND, OR, and NOT.

In a Boolean system, the result of a search is a set of document identifiers. The AND operator narrows retrieval by returning only documents containing all specified terms. The OR operator broadens retrieval by returning documents containing any of the specified terms. The NOT operator excludes documents containing a designated term. While Boolean systems are widely used because of their precision and ease of use, they have a significant limitation: there is no mechanism for ranking results. Either a document satisfies the query completely, or it does not.

6.5.3 Ranked Retrieval and the Vector Space Model

In ranked retrieval, each document is assigned a numerical score representing its degree of similarity to the query, and documents are returned in decreasing order of that score. "According to Panigrahi (2016), since relevance or adequacy is a matter of degree, many information retrieval systems including most Web search engines rank the retrieved results by a score of expected relevance."

In the vector space model, the similarity between a document and a query is computed as the cosine of the angle between their corresponding term vectors. A cosine value close to 1 indicates high similarity; a value close to 0 indicates little or no match. For a query such as "Study of concept analysis in information retrieval," each term's contribution to a document's score is a product of three factors: the weight of the query term (its significance to the user), the term frequency (tf) reflecting the number of occurrences of the term in the document, and the inverse document frequency (idf), which is measured on a logarithmic scale and penalizes terms that appear in a large proportion of the collection.

The cosine similarity measure is preferred over Euclidean distance because it measures the angle between two vectors rather than their absolute distance, making it invariant to the length of documents. This is important because a longer document containing the same ideas as a shorter one should not be penalized for its length.

6.5.4 Probabilistic Retrieval

Another approach to inexact matching is the probabilistic retrieval model, which estimates the probability that a given document is relevant to a query. Bayesian approaches are particularly prominent. "According to Kowalski (2002), a Bayesian network is a directed acyclic graph in which each node represents a random variable and the arcs between nodes represent a probabilistic dependence, and the goal is to calculate the posterior probability that an item contains a given concept given the presence of specific features." This model allows the system to rank documents according to their estimated probability of relevance, providing a principled probabilistic alternative to the geometric vector approach.

Check Your Progress 2

Q1. What is the fundamental difference between exact match and ranked retrieval in IR systems?

Q2. Fill in the blank: In the vector space model, the similarity between a query and a document is typically measured using similarity.

Q3. Name the three Boolean operators commonly used in exact match IR systems.

Q4. True or False: In exact match Boolean retrieval, results are returned in order of relevance.

Q5. According to the vector space model, what does a cosine value close to 0 between a query vector and a document vector indicate?

6.6 RELATIONAL STATEMENTS IN QUERY SPECIFICATION

6.6.1 Structure of Relational Statements

Relational statements are the building blocks of queries in IR systems. They specify the characteristics that records in a retrieved set must possess, or that distinguish a subset of the database. "According to Barve (2016), sets are defined by specifying one or more combinations of an attribute, a relationship, and a value, for example, publication date = 2000, author = Croft, B., or salary > 10,000."

A relational statement thus takes the general form: [attribute] [relational operator] [value]. In an IR context, the attribute may be a bibliographic field such as author, title, publication date, or subject; the relational operator specifies the relationship between the attribute and the value; and the value specifies the target of the search.

6.6.2 Relational Operators

Relational operators are the symbols that establish the relationship between two entities in a query statement. "According to Barve (2016), the symbols =, >, <, >=, <=, and != are called relational operators which establish relation between two entities, and all relational operators have equal preference but they are lower than those of arithmetic operators."

The equality operator (=) is used when the user requires an exact match on a field value, for example, subject = "library science." Inequality operators (<, >, <=, >=) allow range-based queries, such as date > 10/10/2012, which retrieves all documents published after that date. The not-equal operator (!=) retrieves records that do not match a given value.

These operators provide users with flexible and precise control over the retrieval process. They are especially important in hybrid IR systems that combine bibliographic and full-text retrieval, where users may need to filter by structured fields while simultaneously searching unstructured text. Relational operators also play a key role in clustering similar information, as they help identify whether a particular element belongs to a retrieved set.

6.6.3 Sets and Subsets in Query Processing

Most IR systems are designed with the expectation that users will make multiple iterations of query refinement, and they therefore maintain sets and subsets corresponding to each query or component statement. "According to Barve (2016), a set is generally a list of identifying accession numbers of the retrieved records

satisfying the query or component statement, and the user is informed only of the set number assigned by the IRS and the number of records found in the set."

Relational statements define the composition of individual sets. Multiple sets can then be combined using Boolean operators to form more complex queries. For example, Set 1 might be defined by the relational statement date = 2020 and Set 2 by subject = "information retrieval"; the intersection of these two sets, defined as Set 1 AND Set 2, would return all documents on information retrieval published in the year 2020. This modular architecture allows users to build complex retrieval strategies incrementally, testing intermediate results at each step.

6.6.4 Extended and Proximity Operators

Beyond the standard Boolean and relational operators, many IR systems incorporate additional operators to refine query specification. Proximity operators allow users to specify that query terms must appear within a defined distance of one another in the document, such as within the same sentence or paragraph. Truncation operators, represented by wildcard characters such as the asterisk (*), enable prefix searching so that "comput*" would retrieve documents containing "computer," "computing," "computation," and related forms. These extended operators address some of the limitations of standard Boolean queries and increase the flexibility of the query specification process.

Check Your Progress 3

Q1. What are the three components of a relational statement in query specification?

Q2. Match the following operators to their functions:
(a) = (b) > (c) != (d) <=

(i) Retrieves records where the attribute is not equal to the specified value.

(ii) Retrieves records where the attribute value is greater than the specified value.

(ii) Retrieves records matching the exact value specified.
(iii) Retrieves records where the attribute value is less than or equal to the specified value.

Q3. Fill in the blank: A in an IR system is a list of identifying accession numbers of retrieved records satisfying a query or component statement.

Q4. What does the truncation operator (asterisk) do in a query?

6.7RELEVANCE FEEDBACK

6.7.1 Concept and Purpose

Relevance feedback is an iterative process through which users' assessments of retrieved documents are incorporated into the retrieval system to improve the quality of subsequent searches. It is grounded in the recognition that it is often easier for a user to evaluate the relevance of a document once they have seen it than to formulate a precise query from scratch.

"According to Panigrahi (2016), once the user has evaluated the significance of a few items found, the query can be made better; the system can thereby provide assistance for users in enriching the query by displaying a list of features found in many relevant items and another list from irrelevant items." In some implementations, the system automatically improves the query by identifying features that distinguish relevant from irrelevant items and are therefore good predictors of relevance.

6.7.2 Types of Relevance Feedback

Relevance feedback can be implemented in three broad forms: explicit feedback, implicit feedback, and pseudo (blind) feedback.

In explicit relevance feedback, the user directly judges each retrieved document as relevant or non-relevant. The system then uses these judgments to modify the query for subsequent retrieval rounds. This approach is the most accurate but also the most demanding of the user.

In implicit relevance feedback, the system infers user preferences from behavioural signals such as which documents the user clicks on, how long they spend reading a document, or which documents they save or print. This approach is less intrusive but potentially less reliable, as behavioural signals may not accurately reflect relevance judgments.

In pseudo-relevance feedback, the system assumes that the top-ranked documents from an initial retrieval are relevant, and uses them to expand the query without requiring any explicit user input. While this reduces user burden, it can propagate errors if the initial results are poor.

6.7.3 The Rocchio Algorithm

The most influential formal method for implementing relevance feedback is the Rocchio algorithm, developed as part of the SMART IR system in the early 1970s. The algorithm operates within the vector space model and aims to compute an optimal new query vector that maximizes the separation between the centroid (average) of the relevant document vectors and the centroid of the non-relevant document vectors.
The Rocchio formula for the updated query vector is:

$$q_m = \alpha q_0 + \beta \frac{1}{|\ D_r\ |} \sum_{d_j \in D_r} d_j - \gamma \frac{1}{|\ D_{nr}|} \sum_{d_j \in D_{nr}} d_j$$

where $q_m$is the modified query vector, $q_0$is the original query vector, $D_r$is the set of relevant documents, $D_{nr}$is the set of non-relevant documents, and α, β, $\gamma$are weighting parameters. The positive feedback term moves the query vector closer to the relevant documents, while the negative feedback term moves it away from non-relevant documents.

In practice, positive feedback (from relevant documents) is considerably more valuable than negative feedback. Most implementations therefore set the negative feedback weight (γ) lower than the positive feedback weight (β), or even to zero. The algorithm has been shown to be particularly effective for increasing recall by bringing additional relevant documents to the user's attention in subsequent retrieval rounds.

6.7.4 Relevance Feedback and Retrieval Performance

The use of relevance feedback has consistently been shown to improve retrieval performance as measured by both precision and recall. If there exist some relevant documents, the system can use them as a training set to build a classifier with two classes: relevant and not relevant, which will lead to improved measures of recall and precision. Recall is defined as the ratio of relevant documents retrieved over all relevant documents in the collection, and precision is the ratio of relevant documents retrieved over all documents retrieved.

"According to Kowalski (2002), relevance is a matter of degree; some documents are highly relevant and indispensable for the user as they serve the purpose of the user's need, while others may not contribute much to the user's requirements." Relevance feedback exploits this graded nature of relevance by drawing the query progressively toward the region of the document space most densely populated with relevant items.

Interactive relevance feedback also serves a cognitive function by helping users articulate their information needs more precisely over the course of a search session. As users review results and provide judgments, they simultaneously refine their own understanding of what they are seeking, making subsequent queries both more precise and more productive.

Check Your Progress 4

Q1. What is the primary purpose of relevance feedback in an IR system?

Q2. Fill in the blank: In the Rocchio algorithm, the modified query vector is computed by moving the query closer to the centroid of documents and away from documents.

Q3. True or False: In the Rocchio algorithm, negative feedback (from non-relevant documents) is typically weighted more heavily than positive feedback.

6.8 LET US SUM UP

The unit provides a systematic examination of the core processes underlying indexing in information retrieval systems. We have learned how documents are represented for retrieval, covering both manual and automatic indexing techniques, including stop word removal, stemming, and term weighting methods such as TF-IDF and the vector space model. It then explores query formulation, explaining how a user's information need is translated into a formal query using procedural or natural language, with support from thesauri and query expansion strategies.

The unit proceeds to query matching, contrasting exact match Boolean retrieval with ranked retrieval methods based on cosine similarity and probabilistic models. Relational statements and their role in defining sets and subsets through relational operators are examined next, highlighting how structured query conditions are constructed and combined. Finally, the unit addresses relevance feedback, detailing its three principal types and the Rocchio algorithm, demonstrating how iterative user judgments progressively refine retrieval performance and improve both precision and recall.

6.9 ANSWER TO CHECK YOUR PROGRESS

Check Your Progress 1

Ans to Q1: Subject indexing identifies and describes the subject of documents to indicate their content, summarize it, and increase their findability within a collection.

Ans to Q2: Stemming.

Ans to Q3: Term Frequency-Inverse Document Frequency (TF-IDF).

Check Your Progress 2

Ans to Q1: Exact match retrieval returns only documents that fully satisfy all specified query conditions, with no ranking. Ranked retrieval assigns each document a relevance score and returns documents in decreasing order of that score, acknowledging that relevance is a matter of degree.

Ans to Q2: Cosine.

Ans to Q3: AND, OR, and NOT.

Ans to Q4: False. Boolean retrieval returns all documents satisfying the query but does not rank them.

Ans to Q5: A cosine value close to 0 indicates that the document and query vectors are nearly orthogonal, meaning the document has little or no match with the query.

Check Your Progress 3

Ans to Q1: The three components are: an attribute (such as author or date), a relational operator (such as =, >, <), and a value (such as the author's name or a specific date).

Ans to Q2: (a) - (iii); (b) - (ii); (c) - (i); (d) - (iv).

Ans to Q3: Set.

Ans to Q4: The truncation operator retrieves all terms that begin with the root or stem word preceding the asterisk. For example, "Freeze*" would return "freezer," "freezing," and related forms.

Check Your Progress 4

Ans to Q1: The primary purpose of relevance feedback is to use users' assessments of retrieved documents (relevant or non-relevant) to iteratively improve subsequent queries and enhance retrieval performance.

Ans to Q2: Relevant; non-relevant.

Ans to Q3: False. Positive feedback from relevant documents is generally more valuable, and most systems weight it more heavily than negative feedback.

6.10 MODEL QUESTIONS

Short Questions

1. What is an inverted file and what role does it play in IR systems?
2. Distinguish between weighted and unweighted indexing systems.
3. What is meant by query formulation? How does it differ from the user's information need?
4. What are the three Boolean operators used in query specification? State the function of each.
5. Define a relational operator and give two examples of its use in query specification.
6. Explain the concept of a "set" in the context of IR query processing.
7. What is relevance feedback? State the three types with a brief explanation of each.
8. State the role of IDF in the TF-IDF weighting scheme.
9. What is the purpose of stop word removal in the indexing process?

Long Answer Questions

1. Discuss the process of automatic indexing in information retrieval systems. Explain the steps involvedand analyze the advantages of automatic indexing over manual indexing.

2. Explain the vector space model of document representation and query matching.
3. Describe the role of relational statements and Boolean query logic in information retrieval.
4. Explain the concept of relevance feedback in information retrieval.

6.11 REFERENCES AND FURTHER READINGS

References

- Barve, S. (2016). Querying of the Information Retrieval System. In *e-PG Pathshala, Library and Information Science*. Inflibnet Centre.
- Baeza-Yates, R., and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. New York: ACM Press.
- Chowdhury, G. G. (2010). *Introduction to Modern Information Retrieval* (3rd ed.). London: Facet Publishing.
- Kowalski, G. J. (2002). *Information Storage and Retrieval Systems: Theory and Implementation* (2nd ed.). Boston: Kluwer Academic Publishers.Information-Storage-And-Retrieval-Systems-Theory-And-Impl-2e_Kowalski-GJ-2002.pdf
- Larsen, K. S., Schmidt, E. M., and Schwartzbach, M. I. (1989). A Universal Relational Operator. Aarhus, Denmark: Aarhus University, Computer Science Department.
- Lancaster, F. W. (1979). *Information Retrieval Systems: Characteristics, Testing, and Evaluation* (2nd ed.). New York: John Wiley.
- Meadow, C. T., Boyce, B. R., Kraft, D. H., and Barry, C. (2007). *Text Information Retrieval Systems* (3rd ed.). London: Academic Press.
- Panigrahi, P. K. (2016). Basic Concepts and Components of Information Retrieval Systems. In *e-PG Pathshala, Library and Information Science*. Inflibnet Centre.
- Rocchio, J. J. (1971). Relevance Feedback in Information Retrieval. In G. Salton (Ed.), *The SMART Retrieval System: Experiments in Automatic Document Processing* (pp. 313-323). Englewood Cliffs, NJ: Prentice-Hall.cmu+1
- Salton, G., and McGill, M. J. (1983). *Introduction to Modern Information Retrieval*. New York: McGraw-Hill.Information-Storage-And-Retrieval-Systems-Theory-And-Impl-2e_Kowalski-GJ-2002.pdf
- Sparck Jones, K., and Willett, P. (Eds.). (1997). *Readings in Information Retrieval*. San Francisco: Morgan Kaufmann.
- Van Rijsbergen, C. J. (1979). *Information Retrieval* (2nd ed.). London: Butterworth.

- Wolfram, D. (2003). *Applied Informetrics for Information Retrieval Research*. Westport, CT: Libraries Unlimited.

Further Readings

- Manning, C. D., Raghavan, P., and Schutze, H. (2008). *Introduction to Information Retrieval*. Cambridge: Cambridge University Press. (Chapters 6, 9, and 16 are especially relevant for vector space models, scoring, and relevance feedback.)
- Salton, G. (1975). *A Theory of Indexing*. Philadelphia: SIAM.
- Korfhage, R. R. (1997). *Information Storage and Retrieval*. New York: Wiley Computer Publishing.
- Voorhees, E., and Harman, D. (Eds.). (2005). *Text Retrieval Conference*. Cambridge, MA: MIT Press.
- Zadeh, L. A. (1965). Fuzzy Sets. *Information and Control*, 8(3), 338-353.
- Belkin, N. J. (1980). Anomalous States of Knowledge as a Basis for Information Retrieval. *Canadian Journal of Information Science*, 5, 133-143.

*** ***** ***

UNIT 7: EVALUATION AND MEASUREMENT OF IR SYSTEMS

UNIT STRUCTURE

7.1 Learning Objective

7.2 Introduction

7.3 Need for Evaluation

7.4 Evaluation Criteria and Approaches

7.5 Evaluation of Outcome

- 7.5.1 Recall and Precision
- 7.5.2 Fallout and Generality
- 7.5.3 Limitations of Recall and Precision

7.6 Evaluation Experiments

- 7.6.1 Cranfield Tests
- 7.6.2 MEDLARS Evaluation
- 7.6.3 The SMART Retrieval Experiment

7.7 Let Us Sum Up

7.8 Check Your Progress

7.9 Model Questions

7.10 References and Further Readings

7.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- Understand the rationale and purpose of evaluating information retrieval systems.
- Identify and explain the major criteria used in evaluating IR systems.
- Define and calculate recall, precision, fallout, and generality measures.
- Recognise the inherent limitations of recall and precision as evaluation tools.
- Describe landmark evaluation experiments, namely the Cranfield tests, MEDLARS, and the SMART retrieval experiment.

7.2 INTRODUCTION

Evaluation is a systematic process of gathering feedback against an investment of time, energy, money, and intellectual resources. In the context of information retrieval

(IR), evaluation results in indicators that gauge the usefulness of systems and services. According to Shewale in *Evaluation and Measurement of Information Retrieval Systems*, "Information storage and retrieval systems are evaluated from viewpoints such as users, economy, coverage, hardware, software, manpower, and environmental conditions."

The need to evaluate IR systems arises because no system, however sophisticated, can be assumed to perform optimally without empirical verification. Evaluation enables system designers to identify weaknesses, benchmark performance, and guide future development. This unit examines the theoretical foundations of IR evaluation, the key metrics used, and the major evaluation experiments that have shaped the discipline.

7.3 NEED FOR EVALUATION

Evaluation studies in information retrieval investigate the degree to which the stated goals or performance expectations of a system have been achieved. According to Keen (1971), as cited in Shewale's module, there are three major purposes for evaluating an IR system:

1. The need for measures with which to make merit comparisons within a single test situation, i.e., to compare the merits or demerits of two or more systems.
2. The need for measures with which to make comparisons between results obtained in different test situations.
3. The need for assessing the merit of a real-life operational system.

Swanson (1971) extends this framework by noting that evaluation studies may serve one or more of the following purposes:

- To assess a set of goals, a program plan, or a design prior to implementation.
- To determine whether, and how well, goals or performance expectations are being fulfilled.
- To determine specific reasons for successes and failures.
- To uncover principles underlying a successful program.
- To explore techniques for increasing program effectiveness.
- To establish a foundation for further research on the reasons for the relative success of alternative techniques.
- To improve the means employed for attaining objectives, or to redefine subgoals in view of research findings.

An important distinction must be drawn between management-oriented and user-oriented evaluation. When an evaluation is conducted from a managerial point of view, it focuses on administrative and operational efficiency. When conducted from the user's point of view, it is concerned with how well the system serves the needs of its users. Many information scientists advocate that IR system evaluation should always be user-

oriented. According to Cleverdon (1962), as cited by Shewale, "A user-oriented evaluation should try to answer the following questions: To what extent does the system meet both the expressed and latent needs of its user community? What are the reasons for the failure of the system to meet the users' needs? What is the cost-effectiveness of the searches made by the users themselves as against those made by the intermediaries? What basic changes are required to improve the output?"

The economic dimension of evaluation is equally significant. According to Lancaster (1979), the major cost factors to consider include the cost incurred per search, the user's effort in learning to use the system, the effort involved in document delivery, and the time elapsed from query submission to the actual retrieval of information.

7.4 EVALUATION CRITERIA AND APPROACHES

Two broad perspectives govern the evaluation of IR systems: system-oriented and user-oriented. A system-oriented approach focuses on measurable outputs such as speed, coverage, and retrieval ratios, while a user-oriented approach centres on the satisfaction of user information needs.

Cleverdon (1966) identified six criteria for evaluating an IR system:

1. **Recall:** The ability of the system to present all the relevant items.
2. **Precision:** The ability of the system to present only those items that are relevant.
3. **Time lag:** The average interval between when a search request is made and when an answer is provided.
4. **Effort:** The intellectual and physical effort required from the user in obtaining answers to search requests.
5. **Form of presentation:** The manner in which search output is presented, which affects the user's ability to make use of the retrieved items.
6. **Coverage:** The extent to which the system includes relevant material in the collection.

Vickery (1970) organised these criteria into two sets. The first set, relating to the availability of information, includes coverage, recall, and response time. The second set, relating to the selectivity of output, includes precision, usability, and presentation.

Lancaster (1971) similarly proposed five evaluation criteria: coverage of the system, the ability to retrieve wanted items (recall), the ability to avoid retrieval of unwanted items (precision), response time, and the amount of effort required by the user.

Salton and McGill (1983) provided a detailed mapping between these evaluation criteria and the relevant system parameters. They demonstrated that recall and precision are closely associated with indexing exhaustivity, term specificity, the nature of the indexing language, query formulation, and search strategy. Response time depends on the organisation of stored documents, query type, and the size of the

collection. The form of presentation is shaped by the type of display device and the nature of output (bibliographic reference, abstract, or full text).

The success of an information system is, by nature, subjective and depends on what information is needed and how much overhead the user is willing to accept. According to Kowalski (2002), "Overhead can be expressed as the time a user spends in all of the steps leading to reading an item containing the needed information, e.g., query generation, query execution, scanning results of query to select items to read, and reading non-relevant items."

Check Your Progress 1

Q1. Who identified six criteria for evaluating an IR system in 1966?

Q2. Match the following:

Column A	Column B
(a) Recall	(i) Ability to avoid non-relevant items
(b) Precision	(ii) Time between query and result
(c) Time lag	(iii) Ability to retrieve all relevant items
(d) Coverage	(iv) Extent of relevant material in the collection

Q3. Fill in the blank: According to Vickery (1970), the three criteria related to the availability of information are coverage,____, and response time.

Q4. What is the difference between a management-oriented and a user-oriented evaluation?

7.5 Evaluation of Outcome

7.5.1 Recall and Precision

Among the performance criteria identified by Cleverdon, Lancaster, and others, recall and precision are considered the two most fundamental outcome measures. They cannot be measured as readily as criteria such as coverage or form of presentation, which are often determined by policy decisions. Their calculation requires relevance judgements, making them more challenging but also more informative.

According to Kowalski (2002), "When a user decides to issue a search looking for information on a topic, the total database is logically divided into four segments: relevant items retrieved, relevant items not retrieved, non-relevant items retrieved, and non-relevant items not retrieved." This four-cell structure forms the basis for calculating recall and precision.

The standard formulae for recall and precision are:

Recall = (Number of relevant items retrieved / Total number of relevant items in the collection) x 100

Precision = (Number of relevant items retrieved / Total number of items retrieved) x 100

Lancaster's retrieval matrix provides a clear representation of these relationships. If a system retrieves (a + b) documents, where *a* represents relevant documents retrieved (hits) and *b* represents non-relevant documents retrieved (noise), and the remaining (c + d) documents are not retrieved, where *c* represents relevant documents not retrieved (misses) and *d* represents non-relevant documents correctly rejected, then:

- Recall (R) = a / (a + c) x 100
- Precision (P) = a / (a + b) x 100

The ideal system would achieve 100% recall and 100% precision simultaneously. However, this is not achievable in practice. As the level of recall increases, precision tends to decrease, and vice versa. Recall and precision are thus inversely related. According to Kowalski (2002), "Precision starts off at 100 per cent and maintains that value as long as relevant items are retrieved. Recall starts off close to zero and increases as long as relevant items are retrieved until all possible relevant items have been retrieved."

Kowalski (2002) further explains the practical implications of precision-recall graphs: "Assume that there are 100 relevant items in the database and from the graph at precision of .3 (i.e., 30 per cent) there is an associated recall of .5 (i.e., 50 per cent). This means there would be 50 relevant items in the Hit file from the recall value. A precision of 30 per cent means the user would likely review 167 items to find the 50 relevant items."

It is worth noting that Fugmann (1993) questioned the universality of the inverse relationship between recall and precision. Through several examples, he demonstrated that an increase in precision is by no means always accompanied by a corresponding decrease in recall, and that an increase in recall is not always observed to produce a decrease in precision.

Van Rijsbergen proposed a combined effectiveness measure, E, which is a weighted combination of precision and recall. The lower the E value, the greater the effectiveness. The formula is:

$E = 100 \times [1 - (1 + \beta^2) PR / (\beta^2 P + R)]$

where β reflects the relative importance of recall and precision to the user. When β = 0.5, recall is assigned half as much importance as precision.

7.5.2 Fallout and Generality

While conducting a search, it is quite likely that some non-relevant items will be retrieved. This proportion is termed the **fallout ratio**. Simultaneously, the proportion of relevant documents in the collection for a given query is referred to as the **generality ratio**.

Salton (1971) defined these measures formally. The four principal retrieval measures are presented in the table below:

Symbol	Measure	Formula	Explanation
R	Recall	a / (a + c)	Proportion of relevant items retrieved
P	Precision	a / (a + b)	Proportion of retrieved items that are relevant
F	Fallout	b / (b + d)	Proportion of non-relevant items retrieved
G	Generality	(a + c) / (a + b + c + d)	Proportion of relevant items per query

According to Kowalski (2002), "Fallout can be viewed as the inverse of recall and will never encounter the situation of 0/0 unless all the items in the database are relevant to the search." Fallout is, however, less sensitive to changes in the retrieval process than precision because its large denominator, which includes all non-relevant items in the collection, means significant changes in retrieved items are required before the fallout value changes perceptibly.
From a system perspective, the ideal outcome is maximum recall and minimum fallout. This combination implicitly ensures maximum precision as well.

7.5.3 Limitations of Recall and Precision

Despite their widespread use, recall and precision are subject to several significant limitations:

1. **Different users require different levels of recall.** A researcher preparing a comprehensive state-of-the-art report would require high recall, while a user seeking a brief overview of a topic may prefer fewer, highly relevant documents. Users are, however, often unable to specify exactly how many items they wish to retrieve.
2. **Recall assumes equal value of all relevant items.** In practice, retrieved items may have different degrees of relevance, and these degrees may vary from user to user and even from one point in time to another for the same user.
3. **Both recall and precision depend on subjective relevance judgements.** The judgement of whether a document is relevant is inherently subjective. Different users assessing the same document may reach different conclusions.
4. **Recall is non-calculable in operational systems.** Kowalski (2002) notes that "Recall is a very useful concept, but due to the denominator, is non-calculable in operational systems. If the system knew the total set of relevant items in the database, it would have retrieved them." In large, operational databases, determining the total number of relevant items is practically impossible.
5. **Mathematical ambiguities arise in edge cases.** According to Kowalski (2002), precision and recall "encounter mathematical ambiguities and a lack of parallelism between their properties." If there are no relevant items in the database or if no items are retrieved, the mathematical formula becomes 0/0.

6. **Recall is unaffected by retrieval of non-relevant items.** This creates a lack of parallelism between recall and precision, since retrieving more non-relevant items reduces precision but leaves recall unchanged.

Check Your Progress 2

Q1. What does the fallout ratio measure?

Q2. Fill in the blank: Van Rijsbergen's effectiveness measure E combines and into a single score.

Q3. True or False: Recall can be easily calculated in large, operational IR systems.

Q4. Match the following:

Column A	Column B
(a) Precision	(i) Proportion of relevant items per query
(b) Fallout	(ii) Proportion of retrieved items that are relevant
(c) Generality	(iii) Proportion of non-relevant items retrieved
(d) Recall	(iv) Proportion of relevant items retrieved

7.6 EVALUATION EXPERIMENTS

The theoretical measures of recall, precision, fallout, and generality must be tested in controlled experimental settings to yield meaningful results. Over the decades, several landmark experiments have been conducted to evaluate the performance of IR systems. Lancaster (1979) identifies five major steps involved in such evaluation: designing the scope of evaluation, designing the evaluation programme, executing the evaluation, analysing and interpreting results, and modifying the system in the light of evaluation results.

7.6.1 Cranfield Tests

The first large-scale, systematic evaluation of information retrieval systems was undertaken at the College of Aeronautics, Cranfield, United Kingdom, under the direction of C.W. Cleverdon. The initiative is collectively known as the Cranfield experiments and comprises two phases.

Cranfield I (1957-1962)

The first Cranfield study began in 1957 and was reported by Cleverdon in 1962. The project was designed to compare the effectiveness of four indexing systems:

1. An alphabetical subject catalogue based on a subject heading list.

2. A UDC (Universal Decimal Classification) classified catalogue with an alphabetical chain index.
3. A catalogue based on a faceted classification and an alphabetical index to the class headings.
4. A catalogue compiled by the uniterm coordinate index system.

The study involved 18,000 indexed items and 1,200 search topics. Documents were drawn equally from the general field of aeronautics and the specialised field of high-speed aerodynamics. Three indexers were selected: one with subject knowledge, one with indexing experience, and one straight from library school with neither subject background nor indexing experience. Each indexer indexed each source document five times, spending 2, 4, 8, 12, and 16 minutes per document. Each of the 6,000 items so produced was tested in three phases, yielding 18,000 indexed items in total.

The results of Cranfield I were significant and, in several ways, contradicted prevailing assumptions about IR systems:

- Indexing times over four minutes gave no real improvement in performance.
- High-quality indexing could be obtained from non-technical indexers.
- The system operated at a recall rate of 70-90% and a precision rate of 8-20%.
- A 1% improvement in precision could be achieved at the cost of a 3% loss in recall.
- Recall and precision were found to be inversely related.
- All four indexing methods gave broadly similar performance.

According to Kowalski (2002), "One of the first test databases was associated with the Cranfield I and II tests (Cleverdon-62, Cleverdon-66). It contained 1400 documents and 225 queries. It became one of the standard test sets and has been used by a large number of researchers."

Cranfield II

The second Cranfield test was a controlled experiment assessing the effects of components of index languages on the performance of retrieval systems. This study tested altogether 29 index languages formed by the combination of concepts on a collection of 1,400 reports and articles from the field of high-speed aerodynamics and aircraft structures. The objective was to assess the effect of varying each component while keeping the others constant.

The Cranfield experiments are historically significant not only for their findings but also for the methodologies they developed. For the first time, recall and precision were established as the two most important parameters for determining the performance of IR systems, and the inverse relationship between them was empirically demonstrated.

7.6.2 MEDLARS Evaluation

The MEDLARS (Medical Literature Analysis and Retrieval System) evaluation was conducted by the US National Library of Medicine during the period August 1966 to July 1967. The test was conducted on the operational database of MEDLARS, a database of biomedical articles with index entries drawn from MeSH (Medical Subject Headings). The document collection available at the time of the test consisted of approximately 700,000 items.

The objective of the MEDLARS test was to evaluate the existing system and to identify avenues for improvement. Twenty-one user groups were selected from the user community to supply test questions, cover all kinds of subjects in the requests, and represent all categories of users. These user groups provided 302 search requests.
Each query was formulated in terms of MeSH by the system operator and searches were conducted. After completion of each search, sample output was sent to the users for relevance assessment. Rather than merely bibliographic references, photocopies of the actual articles were supplied for relevance assessment. Users were asked to mark each retrieved item using the following scale:

- H1: of major value
- H2: of minor value
- W1: of no value
- W2: value unknown

Precision was calculated using the formula:

Precision ratio = (H1 + H2) / L x 100

where L is the number of sample items retrieved.

The MEDLARS evaluation is important because it was conducted on a large, operational database in a real-world environment, rather than a controlled laboratory setting. This gave its results a degree of ecological validity that complemented the more controlled Cranfield experiments. It revealed specific reasons for retrieval failures, both at the indexing and searching stages, and provided practical guidance for improving the system.

7.6.3 The SMART Retrieval Experiment

The SMART (System for the Mechanical Analysis and Retrieval of Text) system was designed in 1964 at Cornell University under the direction of Gerard Salton, largely as an experimental tool for evaluating the effectiveness of many different types of document analysis and search procedures. According to Salton (1971), as cited by Shewale, the SMART system functioned by taking documents and search queries posed in English, performing a fully automatic content analysis of texts, matching analysed search statements against the contents of documents, and retrieving stored items most similar to the queries.

A number of methods were adopted for automatic content analysis of documents in SMART, including:

- Word suffix cut-off methods.
- Thesaurus look-up procedures.
- Phrase generation methods.
- Statistical term associations.
- Hierarchical term expansion.

The SMART system generated several evaluation measures that were novel at the time:

- A recall-precision graph reflecting the average precision value at ten discrete recall points, from a recall of 0.1 to a recall of 1.0 in intervals of 0.1.
- Two global measures, known as normalised recall and normalised precision, reflecting the overall performance level of the system.
- Two simplified global measures, known as rank recall and log precision respectively.

The SMART experiments were influential because they introduced a rigorous, reproducible framework for evaluating IR systems through automatic processing. Unlike the Cranfield tests, which relied on manual indexing, the SMART system operated entirely automatically, making it possible to test a large number of retrieval strategies efficiently. The SMART system also contributed significantly to the development of the vector space model of information retrieval, which remains foundational in modern IR research.

Check Your Progress 3

Q1. In which year did the first Cranfield study begin?

Q2. What does MEDLARS stand for?

Q3. Fill in the blank: The SMART system was designed in 1964 at University under Gerard Salton.

Q4. Match the following experiments with their key characteristics:

Column A (Experiment)	Column B (Characteristic)
(a) Cranfield I	(i) Evaluated 700,000 biomedical documents
(b) MEDLARS	(ii) Generated recall-precision graphs at 10 discrete points
(c) SMART	(iii) Compared four indexing systems using 18,000 items

Q5. What was the recall rate and precision rate achieved in the Cranfield I experiment?

7.7 LET US SUM UP

Evaluation studies are indispensable in IR because they provide objective evidence of system performance, reveal weaknesses, and inform improvements. The major criteria for evaluation, as identified by Cleverdon (1966), Lancaster (1971), and Vickery (1970), include recall, precision, time lag, effort, form of presentation, and coverage.

Among these criteria, recall and precision are the most widely used outcome measures. Recall measures the proportion of relevant documents retrieved by the system, while precision measures the proportion of retrieved documents that are relevant. They are typically inversely related, and both depend on subjective relevance judgements. Fallout and generality provide additional dimensions of evaluation, measuring the proportion of non-relevant items retrieved and the proportion of relevant items in the collection respectively.

The landmark Cranfield tests established recall and precision as the standard metrics for IR evaluation and demonstrated that all four indexing methods tested yielded broadly similar performance. The MEDLARS evaluation extended this work to a large, operational biomedical database, providing ecologically valid insights. The SMART experiments introduced a fully automatic framework for IR evaluation and pioneered the recall-precision graph as a standard evaluation tool.

7.8 ANSWERTO CHECK YOUR PROGRESS

Check Your Progress 1

Ans to Q1: Cleverdon (1966) identified the six criteria for evaluating an IR system.

Ans to Q2: (a) - (iii); (b) - (i); (c) - (ii); (d) - (iv).

Ans to Q3: Recall.

Ans to Q4: A management-oriented evaluation is conducted from the administrator's point of view, focusing on operational and economic efficiency. A user-oriented evaluation focuses on how well the system meets the expressed and latent needs of its users.

Check Your Progress 2

Ans to Q1: The fallout ratio measures the proportion of non-relevant items in the collection that are incorrectly retrieved by the system in response to a query.

Ans to Q2: Recall and Precision.

Ans to Q3: False. Recall is non-calculable in operational systems because determining the total number of relevant items in a large database is practically impossible.

Ans to Q4: (a) - (ii); (b) - (iii); (c) - (i); (d) - (iv).

Check Your Progress 3

Ans to Q1: The first Cranfield study began in 1957.

Ans to Q2: MEDLARS stands for Medical Literature Analysis and Retrieval System.

Ans to Q3: Cornell.

Ans to Q4: (a) - (iii); (b) - (i); (c) - (ii).

Ans to Q5: The system operated at a recall rate of 70-90% and a precision rate of 8-20%.

7.9 MODEL QUESTIONS

Short Questions

1. Define recall and precision in the context of information retrieval.
2. What is the fallout ratio? How is it calculated?
3. State any three limitations of recall and precision as evaluation measures.
4. Who directed the Cranfield I experiment, and when was it conducted?
5. What is the generality ratio?
6. What scale was used for relevance assessment in the MEDLARS evaluation?
7. What were the five steps of evaluation identified by Lancaster (1979)?
8. What is the significance of the inverse relationship between recall and precision?

Long Questions

1. Explain in detail the criteria identified by Cleverdon and Lancasterfor evaluating information retrieval systems. How do these criteria relate to system parameters as described by Salton and McGill (1983)?
2. Discuss the concepts of recall, precision, fallout, and generality with the help of Lancaster's retrieval matrix. Analyse the inverse relationship between recall and precision and critically examine the limitations of these measures as discussed in the literature.
3. Write a detailed account of the Cranfield experiments. What were the major findings of Cranfield I, and what aspects of IR evaluation did the Cranfield programme establish for the first time?

4. Describe the MEDLARS evaluation experiment. How was relevance assessed in MEDLARS, and how was precision calculated? What makes this experiment significant in the history of IR evaluation?

7.10 REFERENCES AND FURTHER READINGS

References

- Blair, D.C. and Maron, M.E. "An evaluation of retrieval effectiveness for a full-text document retrieval system." *Communications of the ACM*, 28(3), 1985, pp. 289-299.
- Cleverdon, Cyril W. "The Cranfield tests on index language devices." *Aslib Proceedings*, 19(6), 1967, pp. 173-194.
- Cleverdon, C.W. *Report on the first step of an investigation into comparative efficiency of indexing systems*. Cranfield: College of Aeronautics, 1960.
- Cleverdon, C.W. *Report on the testing and analysis of an investigation into comparative efficiency of indexing systems*. Cranfield: College of Aeronautics, 1962.
- Fugmann, R. *Subject Analysis and Indexing: Theoretical Foundation and Practical Advice*. Frankfurt: Indeks-Verlag, 1993.
- Keen, E.M. "Evaluation parameters." In Salton, G. (ed.), *The SMART Retrieval System: Experiments in Automatic Document Processing*. Englewood Cliffs, New Jersey: Prentice Hall, 1971, pp. 74-111.
- Kowalski, G.J. *Information Storage and Retrieval Systems: Theory and Implementation*, 2nd edn. Boston: Kluwer Academic Publishers, 2002.
- Lancaster, F.W. *Information Retrieval Systems: Characteristics, Testing and Evaluation*, 2nd edn. New York: John Wiley, 1979.
- Salton, G. (ed.) *The SMART Retrieval System: Experiments in Automatic Document Processing*. Englewood Cliffs, New Jersey: Prentice Hall, 1971.
- Salton, G. and McGill, M.J. *Introduction to Modern Information Retrieval*. Auckland: McGraw-Hill, 1983.
- Swanson, D.R. "Some unexplained aspects of the Cranfield tests of indexing language performance." *Library Quarterly*, 41, 1971, pp. 223-228.
- Van Rijsbergen, C.J. *Information Retrieval*, 2nd edn. London: Butterworth, 1979.
- Vickery, B.C. *Techniques of Information Retrieval*. London: Butterworth, 1970.

Further Readings

- Sparck Jones, K. and Willett, P. (eds.) *Readings in Information Retrieval*. San Francisco: Morgan Kaufmann, 1997.
- Baeza-Yates, R. and Ribeiro-Neto, B. *Modern Information Retrieval*. Harlow: Addison Wesley, 1999.
- Harman, D. "Overview of the First Text Retrieval Conference (TREC-1)." In *Proceedings of the First Text Retrieval Conference*, NIST Special Publication 500-207. Washington: NIST, 1993.
- Saracevic, T. "Relevance: A review of the literature and a framework for thinking on the notion in information science." *Journal of the American Society for Information Science and Technology*, 58(13), 2007, pp. 1915-1933.

*** ***** ***

UNIT 8: MULTIMEDIA INFORMATION RETRIEVAL

UNIT STRUCTURE

8.1 Learning Objects
8.2 Introduction
8.3 Meaning and scope of multimedia information retrieval
  8.3.1 Core characteristics
  8.3.2 Illustrative example
8.4 Concepts and challenges in multimedia information retrieval
  8.4.1 Major Challenges in Multimedia Retrieval
  8.4.2 Why multimedia retrieval differs from database lookup
8.5 Text Information Retrieval versus Audio and Music Information Retrieval
  8.5.1 Key differences
  8.5.2 Audio and music retrieval methods
  8.5.3 Challenges specific to audio and music
8.6 Image Information Retrieval
  8.6.1 Approaches to image retrieval
  8.6.2 Common image features
  8.6.3 Query interfaces for images
  8.6.4 Limitations
8.7 Video information retrieval
  8.7.1 Representation of video
  8.7.2 Approaches to video retrieval
  8.7.3 Evaluation and TRECVID
  8.7.4 Challenges in video retrieval
8.8 Integrated analytical perspective
8.9 Let Us Sum Up
8.10 Model Questions
8.11 Reference and Further Readings

8.1 Learning Objectives

Explain the Concept and scope of multimedia information retrieval

- Identify the major challenges involved in retrieval multimedia information.
- Distinguish between audio, image and video retrieval systems.

8.2 INTRODUCTION

Information retrieval has traditionally been associated with text, because textual documents can be tokenized, indexed, searched, ranked, and displayed with comparatively mature methods. As digital communication expanded, however, information objects increasingly began to include speech, music, still images, graphics, and video, making retrieval a multi-modal problem rather than a purely textual one.

"According to Kowalski and Maybury in *Information Retrieval Systems: Theory and Implementation*, an Information Retrieval System is capable of storage, retrieval, and maintenance of information, where information may be composed of text, images, audio, video, and other multimedia objects." This definition is important because it places multimedia retrieval within the broader field of information retrieval rather than outside it.

Multimedia information retrieval is therefore concerned with how users locate relevant information when the information is represented in multiple media forms. A user may wish to retrieve a song by humming part of its melody, find medical images that resemble a given scan, identify a news clip containing a named person, or search a lecture video by spoken content. These examples show that the retrieval target may be semantic, perceptual, temporal, or contextual.

In practical terms, multimedia retrieval includes storage, indexing, feature extraction, query formulation, similarity matching, ranking, browsing, and result presentation for heterogeneous media objects. It also requires methods for linking low-level measurable properties, such as colour or pitch, with high-level human concepts, such as "sunset," "sad music," or "political debate."

8.3 MEANING AND SCOPE OF MULTIMEDIA INFORMATION RETRIEVAL

"Multimedia information retrieval is associated with storage, indexing, search, and delivery of multimedia data such as images, videos, sounds, 3D graphics, or their combination." This definition highlights two central ideas: first, multimedia retrieval handles more than one mode of information; second, retrieval depends not only on storage but also on effective representation and access methods.

The scope of the field extends beyond simple keyword search over filenames or manually assigned captions. It includes content-based retrieval, where the system derives searchable representations from the media itself, such as colour histograms for images, acoustic descriptors for audio, or key frames and speech transcripts for video. IBM's QBIC project is a classic example of this approach because it supported image queries based on colour, texture, and shape rather than only on manually assigned metadata.

Multimedia information retrieval also includes integrated retrieval across media. A single video item may contain speech, background sounds, moving images, scene transitions, captions, and named entities, all of which can serve as retrieval evidence. Kowalski and Maybury emphasize that a multimedia item may contain multiple tracks of information that are typically correlated by time, and this temporal linkage is a defining feature of multimedia retrieval.

The field is interdisciplinary. It draws from information retrieval, computer vision, speech recognition, music information retrieval, machine learning, human-computer interaction, indexing theory, and database design. As a result, system design is more

complex than in traditional text retrieval, because the representation of information varies greatly across media types.

8.3.1 Core characteristics

The major characteristics of multimedia information retrieval may be summarized as follows:

- It deals with heterogeneous data, including text, speech, music, images, and video.
- It often depends on feature extraction from raw media content.
- It uses similarity matching rather than only exact matching.
- It must support temporal media such as audio and video.
- It faces a semantic gap between machine-detectable features and user intent.

8.3.2 Illustrative example

Consider a digital news archive containing television broadcasts. A user may search for “speeches by a political leader in New York during an election week.” The system can use closed captions or automatic speech recognition for text evidence, named-entity extraction for persons and locations, key frames for visual browsing, and time metadata for date restriction. Broadcast News Navigator, described by Kowalski and Maybury, illustrates this kind of integrated video retrieval environment.

Check Your Progress 1

Q1. Fill in the blank: Multimedia information retrieval handles _____more than one of information.

Q2. Match the following:

(a) QBIC	(ii) Video retrieval evaluation
(b) TRECVID	(i) Query by image content

Q3. Very short answer: Name any two media types studied in multimedia retrieval.

8.4 CONCEPTS AND CHALLENGES IN MULTIMEDIA INFORMATION RETRIEVAL

The central concept in information retrieval is relevance, but relevance becomes more difficult to operationalize in multimedia environments. In text retrieval, words provide a relatively direct index to meaning. In multimedia retrieval, machines often detect low-level features, while users seek higher-level concepts. An image system may measure colour, shape, and texture, but the user may ask for “a peaceful rural landscape.”

This mismatch is often called the semantic gap. The semantic gap refers to the distance between low-level computable descriptors and the rich conceptual interpretations that humans assign to media. Kowalski and Maybury note that image retrieval has often been based on shallow feature analysis such as color, shape, and texture, while deeper semantic analysis remains a continuing challenge.

Another key concept is representation. Textual data can be represented through terms, stems, frequencies, and linguistic structures. Audio may be represented through transcripts, phonetic units, melody contours, or acoustic features. Images may be represented through global features, local descriptors, object regions, and annotations. Video adds temporal segmentation, shots, scenes, motion, and synchronized audio.

A further concept is query formulation. Multimedia systems must support many forms of querying, including keywords, example-based querying, sketch-based input, humming-based input, browsing, faceted filtering, and relevance feedback. The richer the media, the more difficult it becomes to offer interfaces that are both expressive and easy to use.

8.4.1 Major challenges

Several recurring challenges shape this field:

- **Heterogeneity of data:** Text, sound, image, and video require different indexing and retrieval methods.10ch.docx+1
- **Semantic gap:** Computed features do not fully capture human meaning.
- **Time dependence:** Audio and video unfold over time, so indexing and result presentation must preserve temporal structure.
- **Large-scale indexing:** Multimedia files are often large, making storage, processing, and search computationally intensive.
- **Subjectivity of interpretation:** Different users may judge the same image or music clip differently.
- **Query difficulty:** Users may not know how to express their need in terms the system can process.
- **Evaluation complexity:** Effectiveness must often be measured across multiple tasks, media types, and user behaviors. TRECVID was established specifically to advance benchmarked evaluation for video retrieval.

8.4.2 Why multimedia retrieval differs from database lookup

A conventional database query typically seeks exact values in structured fields. Multimedia retrieval often seeks approximate similarity, semantic correspondence, or partial matching in unstructured content. For example, a user asking for “songs similar in mood to this clip” or “images with a red circular object” is not asking for an exact known field value but for a judged resemblance. QBIC explicitly addressed this need by enabling content-based querying over image features such as colour layout, texture, shape, size, orientation, and object position.

8.5 TEXT INFORMATION RETRIEVAL VERSUS AUDIO AND MUSIC INFORMATION RETRIEVAL

Text information retrieval is the most established branch of the field. In text retrieval, documents are usually represented through words or terms, and the query-document relationship is computed through lexical matching, probabilistic weighting, vector representations, proximity relations, and ranking methods. Because words explicitly encode much of the meaning, indexing is comparatively direct.

Audio and music retrieval, by contrast, face more serious representation problems. Speech is sequential and time-dependent, so access to a specific passage in a long recording requires temporal indexing. If speech can be transcribed, then text retrieval methods can be applied to the transcript, but automatic speech recognition introduces errors, and transcripts often lack punctuation, structure, and disambiguating context. The attached module notes that even good transcripts may omit structural cues and struggle with out-of-vocabulary words such as proper nouns.

Kowalski and Maybury similarly describe spoken language audio retrieval as a specialized multimedia retrieval problem in which linguistic content must often be derived from the speech signal before search can occur effectively. As a result, speech retrieval often combines transcription, language modeling, segmentation, and ranking techniques.

Music information retrieval presents an additional difficulty because music often has no explicit lexical content. In such cases, retrieval may depend on melody, rhythm, pitch contour, timbre, tempo, harmony, or metadata such as artist, composer, and title. The attached chapter draft explains that users may query a music system by humming or playing a tune, after which the system performs similarity matching against stored musical representations.

8.5.1 Key differences

Dimension	Text retrieval	Audio and music retrieval
Basic representation	Words, terms, linguistic units	Speech transcripts, acoustic features, melody or rhythm descriptors
Query style	Keywords, phrases, natural language	Keywords, spoken query, humming, sample audio
Structure	Relatively explicit semantic unit	Strong temporal flow; meaning unfolds over time
Matching	Lexical and semantic matching	Acoustic and similarity matching, often approximate
Common problems	Ambiguity, synonymy, polysemy	Recognition errors, segmentation, lack of explicit words, result presentation

8.5.2 Audio and music retrieval methods

The main methods include:

- **Transcript-based retrieval:** Spoken content is converted into text through automatic speech recognition and then searched using text retrieval techniques.
- **Acoustic feature retrieval:** The system matches clips using low-level audio features such as spectral properties or temporal patterns.
- **Query by humming or example:** The user provides a tune or audio sample, and the system searches for similar contour or rhythm.
- **Metadata retrieval:** Search is performed through artist, album, title, performer, or genre fields.

8.5.3 Challenges specific to audio and music

Audio data is not visually inspectable in the same way as text or images, and it must be listened to over time. The attached module correctly notes that presenting many retrieved audio clips simultaneously is not useful, which means interfaces for browsing and skimming become especially important. In speech retrieval, errors in recognition affect downstream retrieval quality, while in music retrieval the absence of a stable verbal representation complicates indexing.

Check Your Progress 2

Q1. Fill in the blank: In spoken audio retrieval, text retrieval can be applied after generation.

Q2. Match the following:

(a) Humming-based query

(b) ASR

(i) Automatic speech recognition

(ii) Music retrieval interface

Q3. Fact-based question: Give one reason why music retrieval is harder than text retrieval.

8.6 IMAGE INFORMATION RETRIEVAL

Image information retrieval is one of the most extensively developed branches of multimedia retrieval. Before content-based approaches became common, most image systems depended on metadata, captions, and manually assigned descriptors. This method remains useful, but it is expensive, slow, and subjective because different indexers may describe the same image differently.

"According to Eakins and Graham in *Context-based Image Retrieval*, image retrieval depends on attributes such as objects, events, locations, and contextual meaning." It draws on this tradition by listing attributes such as colour, texture, shape, object arrangement, event depiction, specific persons or places, and emotional associations.

The development of content-based image retrieval sought to reduce dependence on manual description. IBM's QBIC system is one of the classic milestones in this area. IBM describes QBIC as a project for querying large online image databases using image content, including colour, colour layout, texture, shape, size, orientation, and position of image objects and regions. This shows the transition from text-associated image access to direct feature-based access.

8.6.1 Approaches to image retrieval

Two broad approaches dominate image retrieval:

- **Text or metadata-based retrieval:** Images are retrieved using titles, captions, tags, subject headings, or surrounding text.
- **Content-based image retrieval:** Images are retrieved by matching visual features derived from the images themselves.

In practice, modern systems often combine both. Metadata can help narrow a search domain, while feature-based methods improve retrieval among visually similar items.

8.6.2 Common image features

Important image descriptors include:

- **Colour:** Often represented through histograms or colour layout statistics.
- **Texture:** Useful for identifying surfaces or repeated patterns.
- **Shape:** Important where object outline or form matters.
- **Spatial arrangement:** Supports queries involving relative object positions.
- **Region-based descriptors:** Focus on parts of an image rather than the entire frame.

Acolour histogram records the proportion of pixels of each colour within an image and can support queries such as finding images with similar dominant colours. Although useful, such methods can fail when semanticmeaning depends on context rather than visual statistics alone.

8.6.3 Query interfaces for images

Image retrieval systems commonly provide browsing interfaces, query-by-example interfaces, sketch-based interfaces, and palette-based selection tools. QBIC explicitly supported query by image example and user-drawn input, reflecting the need to let users express visual needs without relying only on words. Browsing and navigation, as well as example-based querying, are two important interface families.

8.6.4 Limitations

Image retrieval remains challenging because similar meanings may have very different visual appearances, while visually similar images may have very different meanings. A red circular object may correspond to a ball, a traffic sign, a flower center,

or a medical feature. Kowalski and Maybury note that shallow feature analyses are useful but insufficient for deeper semantic interpretation.

8.7 Video information retrieval

Video retrieval combines the difficulties of image retrieval and audio retrieval while adding temporal structure, motion, scene transitions, and cross-modal synchronization. A video item is not merely a sequence of still images. It may include changing visual content, spoken dialogue, music, environmental sounds, captions, and metadata, all unfolding over time.

The video shares the visual nature of image data but is also time-dependent like audio, and is often associated with a synchronized audio track. Consequently, video retrieval systems typically apply text, image, and audio techniques together rather than relying on a single method.

8.7.1 Representation of video

Important representational units in video retrieval include:

- **Frames:** Individual still images in the sequence.
- **Shots:** Uninterrupted camera sequences.
- **Scenes:** Larger semantic segments composed of related shots.
- **Key frames:** Representative frames used for indexing and browsing.
- **Speech transcripts or captions:** Textual evidence extracted from audio channels.
- **Named entities and events:** Higher-level semantic annotations useful for topic-based retrieval.

The use of key frames is especially important because users cannot efficiently inspect entire videos during result browsing. Key frames provide compact visual surrogates, while associated speech or caption text can improve textual access.

8.7.2 Approaches to video retrieval

Video retrieval commonly uses the following methods:

- **Metadata indexing:** Retrieval through title, source, date, genre, or creator.
- **Closed-caption or ASR-based retrieval:** Search over spoken content transformed into text.
- **Visual feature analysis:** Search using colour, texture, motion, or shot-level image descriptors.
- **Segmentation and key frame extraction:** Breaking video into shots and indexing representative frames.
- **Named-entity and concept detection:** Identifying people, organizations, places, and events from multimodal evidence.

Kowalski and Maybury discuss Broadcast News Navigator as a system that automatically captures, annotates, segments, summarizes, and visualizes stories from broadcast video using text, speech, and image processing technologies. This is a

significant example because it shows that useful video retrieval often depends on integrated multi-stream analysis rather than a single descriptor type.

8.7.3 Evaluation and TRECVID

Video retrieval research advanced substantially through shared evaluation frameworks. NIST states that TRECVID emerged from the TREC video track in 2001 and 2002 and became an independent evaluation from 2003 onward, with the goal of promoting research in automatic segmentation, indexing, and content-based retrieval of digital video. The broader TRECVID overview further explains that the initiative provides common datasets, tasks, and metrics for benchmarking systems.

This kind of evaluation is essential because video retrieval effectiveness depends on multiple subtasks, such as shot boundary detection, feature extraction, ad hoc search, captioning, and scene understanding. Without common benchmarks, comparing systems would be difficult and often methodologically weak.

8.7.4 Challenges in video retrieval

Key problems include very large file sizes, automatic segmentation errors, noisy transcripts, semantic ambiguity, and the complexity of presenting results. Even when a system retrieves the right video, the user may still need help locating the exact relevant moment. For this reason, temporal indexing and scene-level navigation are central to effective video retrieval.

Check Your Progress 3

Q1. Fill in the blank:

(a) A representative frame used for indexing a video segment is called a frame.

Q2. Match the following:

(a) Shot

(b) TRECVID

(i) Video retrieval benchmark initiative

(ii) Uninterrupted camera sequence

8.8 Integrated analytical perspective

A useful way to understand multimedia information retrieval is to see it as a continuum from explicit symbolic representation to implicit perceptual representation. Text lies closer to explicit symbolic meaning because words are already linguistic units. Audio, image, and video require additional transformation before they become searchable in semantically useful ways.

This does not mean that text retrieval is simple or that multimedia retrieval is only a technical extension of text retrieval. Rather, it means that each medium raises different indexing, matching, and interface problems. Text retrieval is primarily concerned with lexical and semantic access; audio retrieval with temporal acoustic and

linguistic access; image retrieval with visual similarity and semantics; and video retrieval with temporally synchronized multi-modal evidence.

The most effective multimedia systems are therefore hybrid systems. They combine metadata, automatically extracted features, machine learning, and interactive browsing. They also rely increasingly on user feedback, because relevance in multimedia contexts is often subjective and task-dependent.

8.9 LET US SUM UP

Multimedia information retrieval extends the principles of information retrieval to heterogeneous and often time-dependent media such as speech, music, images, and video. Its central tasks include representation, indexing, querying, similarity matching, ranking, browsing, and evaluation.

The field differs from traditional text retrieval because much of multimedia meaning is not directly available in word form. This creates the semantic gap and makes content-based methods necessary, especially for images, music, and video.

Text retrieval remains the most mature branch, while audio and music retrieval depend heavily on transcription, acoustic modeling, and similarity matching. Image retrieval relies on both metadata and content-based features such as color, texture, and shape. Video retrieval combines speech, image, motion, segmentation, and temporal indexing, and has benefited greatly from benchmark initiatives such as TRECVID.

Libraries, archives, surveillance systems, entertainment platforms, educational repositories, news services, and scientific collections all require methods for locating relevant multimedia content accurately and efficiently.

ANSWER TO CHECK YOUR PROGRESS

Check Your Progress 1

Ans to Q1: mode or medium.

Ans to Q2: (a) ii, (b) i.

Ans to Q3: Any two of text, audio, music, image, video, or graphics.

Check Your Progress 2

Ans to Q1: transcript or transcription.

Ans to Q2: (a) ii, (b) i.

Ans to Q3: Because music often lacks explicit word-based content and must be matched through features such as pitch, rhythm, or timbre.

Check Your Progress 3

Ans to Q1: key.10ch.

Ans to Q2: (a) ii, (b) i.

Ans to Q3: Closed captions or automatic speech recognition transcripts.

8.10 Model Questions

Short questions

1. Define multimedia information retrieval.
2. State any two characteristics of multimedia information.
3. What is meant by the semantic gap?
4. Distinguish between metadata-based and content-based image retrieval.
5. What is automatic speech recognition in audio retrieval?
6. What is a key frame in video retrieval?
7. Name any three image features used in content-based retrieval.
8. Why is music retrieval more difficult than text retrieval?
9. What is TRECVID?
10. Mention one example of a multimedia retrieval system.

Long questions

1. Explain the concept, scope, and challenges of multimedia information retrieval with suitable examples.
2. Compare text information retrieval with audio and music information retrieval in terms of representation, indexing, query formulation, and retrieval challenges.
3. Discuss the principles, methods, and limitations of image information retrieval, with reference to content-based image retrieval.
4. Examine the major approaches to video information retrieval and explain the importance of segmentation, key frames, and transcript-based access.
5. Write a critical note on the role of evaluation frameworks in multimedia retrieval research, with special reference to TRECVID.

8.11 REFERENCES AND FURTHER READING

References

- Chaudhary, G. G. *Introduction to Modern Information Retrieval*. 2nd ed. London: Facet Publishing, 2004.
- Downie, J. S. “Music Information Retrieval.” In *Annual Review of Information Science and Technology*, Vol. 37. Information Today, 2002.
- Eakins, J. P., and M. E. Graham. *Context-Based Image Retrieval: A Report to the JISC Technology Applications Programme*. 1999.
- Kowalski, Gerald J., and Mark T. Maybury. *Information Retrieval Systems: Theory and Implementation*. 2nd ed. Boston: Kluwer Academic Publishers, 2002. Information-Storage-And-Retrieval-Systems-Theory-And-Impl-2e Kowalski-GJ-2002.pdf
- Niblack, W. et al. “QBIC Project: Querying Images by Content, Using Color, Texture, and Shape.” IBM Research, based on the 1993 publication record.
- Petkovic, D. et al. “Recent Applications of IBM’s Query by Image Content (QBIC).” IBM Research, 1996.

- National Institute of Standards and Technology. “TREC Video Retrieval Evaluation: TRECVID.” NIST.
- Smeaton, A. F., et al. “An Overview on the Evaluated Video Retrieval Tasks at TRECVID 2022.” arXiv, 2023.
- Tang, N., and J. Furner. “Multimedia Information Retrieval Systems: An Overview.” 1999.

🔍 Further Readings

- Baeza-Yates, R., and B. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, 1999.
- Lancaster, F. W. *Indexing and Abstracting: Theory and Practice*. 3rd ed. London: Facet Publishing, 2003.
- Rasmussen, E. “Libraries and Bibliographic Systems.” In Baeza-Yates and Ribeiro-Neto, *Modern Information Retrieval*. ACM Press, 1999.

*** ***** ***

UNIT 9: USERS OF IR SYSTEMS AND INFORMATION NEEDS

UNIT STRUCTURE

9.1 Learning Objectives
9.2 Introduction
9.3 Users and Their Nature
  9.3.1 Categories of Users
  9.3.2 User Functions in an IR System
9.4 Types of Information Needs
  9.4.1 Conceptual Framework: Need, Want, Demand, and Use
  9.4.2 Psychological Dimensions of Information Need
  9.4.3 Classification of Information Needs
9.5 Information Needs in Different Areas of Activity
  9.5.1 Industrial Information Needs
  9.5.2 Planning Information Needs
  9.5.3 Decision-Making and Information Needs
  9.5.4 Research and Development Information Needs
  9.5.5 Information Needs in Business
9.6 Information Seeking Behaviour of Users
  9.6.1 Models and Processes of Information Seeking
9.7 User Studies and Their Role in IR System Design and Improvement
  9.7.1 Objectives and Need for User Studies
  9.7.2 Methods and Techniques of User Studies
  9.7.3 Data Analysis and Implications for IR Systems
  9.7.4 Limitations of User Studies
9.8 Let Us Sum Up
9.9 Answer to Check Your Progress
9.10 Model Questions
9.11 References and Further Readings

9.1 LEARNING OBJECTIVE

After going through this unit, you will be able to:

- Identify and classify the different categories of users of Information Retrieval Systems based on their nature of work, area of activity, and level of expertise.

- Explain the concept of information need and distinguish between the related terms of need, want, demand, and use with reference to established theoretical frameworks.
- Analyse the specific information needs that arise in distinct areas of professional activity, including industry, planning, decision-making, research and development, and business.
- Describe the sequential stages of information seeking behaviour and explain how an understanding of this behaviour informs the design of user-centred retrieval systems.
- Evaluate the objectives, methods, implications, and limitations of user studies as a systematic methodology for the design and improvement of IR systems.

9.2 INTRODUCTION

An Information Retrieval System (IRS) is a complex arrangement of people, processes, and technologies designed with a single overarching purpose: to connect users with the information they require, as efficiently and accurately as possible. Within this arrangement, the user occupies a central and defining position. Every functional decision in system design, from the choice of indexing vocabulary to the design of query interfaces, ultimately derives its rationale from the characteristics, needs, and behaviours of users. It is, therefore, not sufficient to study an IRS only from a technical standpoint. A thorough understanding of who the users are, what they need, how they seek information, and how their interactions with systems can be systematically studied is indispensable for the development of effective IR environments.

This unit begins by examining the nature and categories of users, proceeds to analyse the concept and classification of information needs, and then explores the specific information requirements that emerge in distinct areas of professional activity. The unit subsequently discusses models of information seeking behaviour and concludes with a focused treatment of user studies as a methodological tool for improving IR system design.

9.3 USERS AND THEIR NATURE

In information science, the term "user" carries a specific and deliberate meaning. "According to Vinit Kumar in 'Users of Information Retrieval Systems'," the person who is actively seeking access to information and who, when successful, obtains and uses the information is described as user". This definition draws attention not merely to the act of searching but to the complete cycle of access, retrieval, and utilisation. In practical contexts, the user may also be referred to as a patron, client, member, or end-user, depending on the institutional setting.

For an IRS to be effective, three fundamental questions about users must be addressed: who the users are and what is the nature of their needs; how they seek the

information they require; and what use patterns they exhibit once information is obtained. Answers to these questions together determine the design parameters and performance benchmarks of any functional retrieval system.

9.3.1 Categories of Users

Users can be categorised on several bases. Broadly, three major groups can be distinguished according to activity:

a) Researchers engaged in basic and applied sciences who require access to the latest scientific and technical literature.
b) Practitioners and technicians involved in developmental and operational activities across fields such as technology, industry, agriculture, medicine, and communication.
c) Managers, planners, and decision-makers who participate in developmental activities in both the public and private sectors.

Beyond activity-based classification, users may also be grouped by nature of work (engineers, scientists, policy-makers), by psychological criteria (varying degrees of motivation and technological confidence), and by level of education and expertise such as expert or novice.

9.3.2 User Functions in an IR System

Within any IRS, users actively perform a sequence of functions constituting the information access process, including searching, browsing, selecting, evaluating, and utilising retrieved information. Users may also engage in query reformulation when initial results are unsatisfactory. Kowalski (2002) notes that users who consult with an intermediary such as a librarian show failure rates as low as 10 percent, compared to 70 percent failure rates on single unaided queries, which underscores the significance of the human element in IR.

9.4 TYPES OF INFORMATION NEEDS

The concept of information need is central to information retrieval. Before a system can be designed to satisfy any need, that need must first be accurately understood and classified. Information need exists at the intersection of cognitive states, practical circumstances, and institutional contexts, and it must therefore be examined with conceptual precision.

9.4.1 Conceptual Framework: Need, Want, Demand, and Use

A foundational contribution to clarifying information need was made by Maurice Line in 1974. "According to Maurice Line in 'Draft Definitions: Information and Library

Needs, Wants, Demands and Uses' (ASLIB Proceedings, 1974)," the terms need, want, demand, and use represent distinct but interrelated states:

- A **need** is what an individual ought to have for work, research, recreation, or personal development. A need may exist even when the individual is unaware of it and constitutes a potential demand.
- A **want** is what an individual would like to have, irrespective of whether it reflects an objective need. A want is also a potential demand.
- A **requirement** covers the combined territory of need and demand and is a broader operationally useful term.
- A **demand** is a formal request for specific information, shaped partly by expectations conditioned by existing service provision. A demand is a potential use.
- A **use** is the actual engagement with information, whether resulting from a deliberate demand, browsing, or accidental encounter.

This framework makes clear that a user may need information without demanding it, demand information without needing it, and use information that satisfies neither.

9.4.2 Psychological Dimensions of Information Need

"According to Robert S. Taylor in 'Question Negotiation and Information Seeking in Libraries' (College and Research Libraries, 1968)," information need can be understood at four progressive psychological levels:

a) **Visceral need**: An actual but unexpressed need, existing at a subconscious level.
b) **Conscious need**: An ill-defined awareness of a knowledge gap, not yet formulated in specific terms.
c) **Formal need**: An area of uncertainty articulated in concrete terms.
d) **Compromised need**: A query reformulated to match the capabilities of the available system.

A closely related contribution was made by N.J. Belkin. "According to N.J. Belkin in 'Anomalous State of Knowledge as a Basis for Information Retrieval' (Canadian Journal of Information Science, 1980)," information need arises when the user recognises an anomaly or gap in their current state of knowledge, and the user is generally unable to precisely specify what information is required to resolve that anomaly. This ASK model has had considerable influence on the design of interactive IR systems.

9.4.3 Classification of Information Needs

At the applied level, information needs are classified into four widely recognised types:

a) **Social or pragmatic information needs**: Information to cope with daily life such as weather, transport schedules, and civic notices.
b) **Recreational information needs**: Information satisfying cultural and leisure interests.
c) **Professional information needs**: Information for effective functioning within a professional domain, including new techniques, standards, and research findings.
d) **Educational information needs**: Information fulfilling academic requirements or supporting the acquisition of new skills.

Information needs can additionally be classified as **kinetic** (immediate and directed at a problem at hand) or **potential** (latent, embedded in a user's professional context, awaiting articulation).

CHECK YOUR PROGRESS – I

Q1. Who coined the distinction between information need, want, demand, and use in the context of information science?

Q2. Match the following:
(i) Visceral need — (a) Formally articulated uncertainty
(ii) Formal need — (b) Unexpressed subconscious need
(iii) Compromised need — (c) Need adjusted to system capabilities

Q3. Fill in the blank: According to N.J. Belkin, information need arises from the user's _ State of Knowledge.

Q4. Which type of information need addresses daily life requirements such as transport timings?

Q5. True or False: A demand always reflects a genuine information need.

9.5 INFORMATION NEEDS IN DIFFERENT AREAS OF ACTIVITY

Professional contexts generate unique and specialised information requirements. As noted by Ms. Pauline Atherton in her "Handbook for Information Systems and Services" (UNESCO, 1977), the information needs of any user group can be ascertained through methods such as the study of organisational charts, questionnaire surveys, interviews at multiple hierarchical levels, analysis of user publications, and direct observation at the workplace.

9.5.1 Industrial Information Needs

According to Kumar (2017), industrial information needs fall into four broad categories:

a) **Technological information**: Data on production processes, materials, machinery, patents, and technical standards.
b) **Company-oriented information**: Details concerning competitor organisations' strategies and performance.

c) **Economic information**: Market trends, pricing, input costs, and macroeconomic conditions.
d) **Policy information**: Regulatory requirements, government policies, and legal frameworks.

New industries additionally require information on feasibility, location planning, raw material availability, labour conditions, financial frameworks, and market strategy.

9.5.2 Planning Information Needs

Planning is inherently information-intensive. According to Kumar (2017), five interactive planning steps each generate specific information requirements:

a) Establishing goals and objectives requires large volumes of data on past and present situations.
b) Identifying necessary activities requires contextual information on each event.
c) Identifying required resources requires information on personnel and material availability.
d) Estimating activity durations requires prior experiential and analytical data.
e) Sequencing activities for optimal results requires analytical information to evaluate pathways.

The future-oriented nature of planning creates demands for IR systems capable of synthesising data across sources in support of forecasting.

9.5.3 Decision-Making and Information Needs

The quality of decisions is generally proportional to the quality and completeness of available information. According to Kumar (2017), three levels of decision-making correspond to three distinct information requirement types:

a) **Strategic decision-making** requires aggregated, cross-functional, forward-looking information for policy formation and long-term resource allocation.
b) **Tactical decision-making** requires specific, quantitative information for medium-term implementation of strategic goals.
c) **Operational decision-making** requires real-time, detailed information on processes and exceptions.

The theorisation of management information systems by Gorry and Morton (1977) has contributed significantly to mapping information needs across these decision-making levels.

9.5.4 Research and Development Information Needs

R&D is among the most information-intensive forms of human activity, encompassing both basic and applied research. According to Kumar (2017),

researchers require information for purposes ranging from problem definition and formulation of solutions to interpreting collected data and integrating findings into the existing body of knowledge. R&D professionals rely on periodicals, technical reports, theses, conference proceedings, and review monographs, and the challenge for IR systems is to support the complex, iterative, and interdisciplinary nature of scientific inquiry.

9.5.5 Information Needs in Business

Business organisations operate within complex economic, legal, political, social, and technological environments. According to Kumar (2017), the principal information needs of business include capital procurement, technical know-how, knowledge of existing policies and regulations, market and competition analysis, foreign trade data, and management information for internal coordination. S.A. Olanigan (1987) observed that business consultants in particular require timely and accurate information across multiple domains simultaneously, placing significant demands on the breadth and currency of available retrieval systems.

CHECK YOUR PROGRESS – II

Q1. Fill in the blank: Industrial information needs are broadly divided into four categories: technological, company-oriented, economic, and _ information.

Q2. Which scholar identified specific methods for ascertaining the information needs of a user community in 1977?

Q3. Match the following:

(i) Strategic decision-making — (a) Resource allocation and standards
(ii) Tactical decision-making — (b) Policy formation and long-term goals

Q4. Name any three document types used by R&D professionals to satisfy their information needs.

Q5. True or False: Business information needs are uniform across all sectors of the economy.

9.6 INFORMATION SEEKING BEHAVIOUR OF USERS

"According to T.D. Wilson in 'On User Studies and Information Needs' (Journal of Documentation, 1981)," information seeking behaviour (ISB) is the attempt of the user to obtain needed information, resulting from the recognition of some need as perceived by the user. ISB may take several forms: directed queries to formal IR systems, manual searches through physical collections, consultation of informal human networks, or opportunistic browsing. Interpersonal information exchange, which involves an element of social reciprocity noted widely in sociological literature, is also a recognised component of ISB. Dissatisfaction with retrieved information can generate new information needs and initiate further rounds of seeking behaviour.

9.6.1 Models and Processes of Information Seeking

According to Girija Kumar (1990), as cited by Vinit Kumar (2017), information seeking behaviour involves the following sequential processes:

(a) Identifying objectives: Establishing a clear understanding of the purpose for which information is required.

(b) Defining needs: Translating objectives into specific, articulable information requirements.

(c) Accessing information systems: Selecting and approaching appropriate IR systems, databases, or human sources.

(d) Establishing sources: Identifying specific documents, datasets, or individuals likely to contain relevant information.

(e) Information acquisition: Retrieving information from identified sources.

(f) Use of information: Applying acquired information to the original problem.

(g) Satisfaction or dissatisfaction: Evaluating the outcome; if unresolved, the process recommences.

ISB integrates three basic resources: people, information, and systems. Kowalski (2002) notes that typical users enter short, imprecise search statements that return excessive data, and that improved query expansion tools, relevance feedback mechanisms, and visualisation capabilities are among the most productive areas for IR system enhancement.

9.7 USER STUDIES AND THEIR ROLE IN IR SYSTEM DESIGN AND IMPROVEMENT

"According to Wilson, user study is a term which covers a very wide range of potential research, from the study of users' choices of books from a university library, through reactions to online search outputs, to the in-depth analysis of the underlying needs that result in information seeking". The Centre for Research on User Studies at the University of Sheffield identified the general objective of user research as a further understanding of the processes of information transfer, expected to lead to the improvement of information transfer systems of all types.

H. Menzel (1966) proposed a tripartite categorisation of user studies:

(a) **Behaviour studies**: Studies of overall user-community interaction patterns with communication systems, without reference to any specific information event.

(b) **Use studies**: Studies focused on the use of specific communication media such as primary periodicals or electronic databases.

(c) **Information flow studies**: Studies examining patterns of information transfer and diffusion across communication systems.

9.7.1 Objectives and Need for User Studies

Based on Singh (2017), user studies pursue several interconnected objectives:

(a) To examine patterns of library and information system use and identify gaps.

(b) To evaluate the relevance and adequacy of available information resources.

(c) To introduce documentation and information services aligned with demonstrated user needs.

(d) To evolve resource-sharing frameworks with other libraries and information centres.

The need for user studies arises from the persistent gap between what users actually require and what systems routinely provide. Singh (2017) identifies specific motivations including the need to identify levels and kinds of user needs, assess the strengths and weaknesses of existing services, identify factors that discourage use of the system, and provide data to improve the organisation of information services at both local and national levels.

9.7.2 Methods and Techniques of User Studies

Several methods are available, grouped into conventional, indirect, and other categories:

General or Conventional Methods:

- **Questionnaire method**: A structured schedule of questions administered to a sample of users; economical and applicable to large populations but susceptible to low response rates.
- **Interview method**: Structured or unstructured personal interaction; yields high response rates and follow-up clarification but is time-consuming.
- **Diary method**: Users maintain detailed records of information activities over time; captures longitudinal patterns but sustaining participation can be difficult.
- **Observation method**: Direct recording of users' behaviour at information access points; yields reliable data on actual behaviour but may influence the observed behaviour.
- **Operations research**: Application of scientific and mathematical methods to information-related decision problems, including resource allocation.

Indirect Methods:

- **Analysis of library records**: Examination of circulation, interlibrary loan, and query logs to infer patterns of need and use.
- **Citation analysis**: Examination of bibliographic citations to determine which document types are actually used by practitioners and researchers.

Other Methods include computer feedback analysis and unconventional methods such as analysis of autobiographical accounts of information experience.

9.7.3 Data Analysis and Implications for IR Systems

Data collected may be analysed through statistical, semantic, psycho-social, or economic techniques. According to Singh (2017), user study findings carry direct implications for IR system improvement:

(a) **Collection development**: Empirical basis for selection and acquisition aligned with demonstrated user needs.

(b) **Weeding of documents**: Identification of unused or outdated materials for systematic deselection.

(c) **Allocation of resources**: User-expressed priorities guide equitable allocation of budgetary and human resources.

(d) **Design of user-based information services**: Only services with demonstrated broad demand can be sustainably justified.

(e) **Improvement of IR techniques**: Studies of search behaviour and query formulation directly inform improvements to indexing, retrieval algorithms, and interface design.

Kowalski (2002) emphasises that advances in query expansion, relevance feedback, and visualisation tools represent the most impactful areas of IR development, and that user studies provide the empirical foundation upon which such developments are validated.

9.7.4 Limitations of User Studies

Despite their importance, user studies have methodological and practical limitations. Singh (2017) identifies frequent shortcomings including imprecisely formulated objectives, biased or unrepresentative sampling, inappropriate statistical analysis methods, and the inherent complexity of information needs, which may resist accurate measurement through standard survey instruments. The composition of the user population is a critical variable, as the information needs of academic users may differ markedly from those in industrial or governmental settings; failure to account for this can produce findings that are not genuinely representative.

CHECK YOUR PROGRESS – III

Q1. Name the three categories of user studies proposed by H. Menzel in 1966.

Q2. Fill in the blank: The Centre for Research on User Studies at the University of _ identified understanding information transfer as the general objective of user research.

Q3. Match the following methods with their characteristics:

(i) Questionnaire method — (a) Economical and suitable for large populations
(ii) Observation method — (b) Records actual behaviour but is time-consuming

Q4. True or False: Citation analysis is classified as a direct method of user study.

Q5. Name any two implications of user studies for improving library and IR systems.

9.8 LET US SUM UP

This unit has examined the user dimension of information retrieval in a systematic and comprehensive manner. Key points addressed include the categorisation of users by activity type, nature of work, and psychological criteria; Line's 1974 conceptual framework distinguishing need, want, demand, and use; Taylor's four levels of information need and Belkin's ASK model; the distinct information requirements of industry, planning, decision-making, R&D, and business; the multi-stage, iterative processes of information seeking behaviour; and user studies as the methodological approach for understanding and improving the human-system interface in IR environments.

9.9 ANSWER TO CHECK YOUR PROGRESS

CHECK YOUR PROGRESS – I

Ans to Q1: Maurice Line, in his 1974 paper "Draft Definitions: Information and Library Needs, Wants, Demands and Uses," published in ASLIB Proceedings.

Ans to Q2: (i) – (b); (ii) – (a); (iii) – (c)

Ans to Q3: Anomalous

Ans to Q4: Social or Pragmatic information need.

Ans to Q5: False. A user may demand information that does not correspond to a genuine need, as Line's framework makes clear.

CHECK YOUR PROGRESS – II

Ans to Q1: Policy

Ans to Q2: Ms. Pauline Atherton, in her Handbook for Information Systems and Services (UNESCO, 1977).

Ans to Q3: (i) – (b); (ii) – (a)

Ans to Q4: Periodicals, technical reports, conference proceedings (theses, monographs, and review articles are also acceptable).

Ans to Q5: False. Business information needs vary considerably by sector, organisational size, and the specific environmental forces impacting a particular enterprise.

CHECK YOUR PROGRESS – III

Ans to Q1: Behaviour studies, use studies, and information flow studies.

Ans to Q2: Sheffield.

Ans to Q3: (i) – (a); (ii) – (b)

Ans to Q4: False. Citation analysis is classified as an indirect method of user study.

Ans to Q5: Any two from: collection development, weeding of documents, allocation of resources, design of user-based information services, improvement of cataloguing and retrieval techniques.

9.9 MODEL QUESTIONS

Short Questions

1. Define the term "user" in the context of Information Retrieval Systems.
2. What is meant by "kinetic" and "potential" information needs?
3. Explain the Anomalous State of Knowledge (ASK) model proposed by N.J. Belkin.
4. Distinguish between "need" and "demand" in the context of information science, with reference to Maurice Line's framework.
5. What are behaviour studies as a category of user research?
6. List any three objectives of conducting user studies.
7. What is citation analysis, and how is it used as an indirect method in user studies?
8. Name the five-step planning process and indicate the type of information required at each stage.

Long Questions

1. Discuss in detail the various types of information needs with examples. How does their classification contribute to the design of effective IR systems?
2. Examine the information needs in research and development activities. What specific information requirements do R&D scientists have, and how can an IRS be designed to meet these needs effectively?
3. Write a detailed account of the information seeking behaviour of users. Discuss any one model of information seeking behaviour with reference to the stages and processes involved.

4. What are user studies? Discuss the major methods and techniques employed in user studies, and explain their implications for the improvement of IR systems and library services.
5. Analyse the relationship between decision-making and information needs. With reference to the different levels of decision-making, explain how information systems can be designed to support each level.

9.9 REFERENCES AND FURTHER READINGS

References

- Atherton, Pauline A. *Handbook for Information Systems and Services*. Paris: UNESCO, 1977.
- Belkin, N.J. "Anomalous State of Knowledge as a Basis for Information Retrieval." *Canadian Journal of Information Science*, 5, 1980.
- Girija Kumar. "Defining the Concept of Information Needs." In Binwal, J.C. et al. *Social Science Information: Problems and Prospects*. New Delhi: Vikas Publishing House, 1990.
- Gorry, G.A. and Morton, M.S. Scott. "A Framework for Management Information Systems." *Sloan Management Review*, 1977.
- Kowalski, G.J. *Information Storage and Retrieval Systems: Theory and Implementation*. 2nd ed. Boston: Kluwer Academic Publishers, 2002.Information-Storage-And-Retrieval-Systems-Theory-And-Impl-2e_Kowalski-GJ-2002.pdf
- Kumar, Vinit. "Users of Information Retrieval Systems." e-PG Pathshala Module 9. INFLIBNET Centre, 2017. https://epgp.inflibnet.ac.in
- Line, M.B. "Draft Definitions: Information and Library Needs, Wants, Demands and Uses." *ASLIB Proceedings*, 26, 1974.
- Menzel, H. "Information Needs and Uses in Science and Technology." *Annual Review of Information Science and Technology*, 1, 1966.
- Olanigan, S.A. "Information Needs of the Consultants to Business Enterprises." *International Library Review*, 19, 1987.
- Singh, Sewa. "User Studies, Users Education." e-PG Pathshala Module 16. INFLIBNET Centre, 2017. https://epgp.inflibnet.ac.in
- Taylor, R.S. "Question Negotiation and Information Seeking in Libraries." *College and Research Libraries*, 29, 1968.
- Wilson, T.D. "On User Studies and Information Needs." *Journal of Documentation*, 37, 1981.
-

Further Readings

- Case, Donald O. *Looking for Information: A Survey of Research on Information Seeking, Needs and Behaviour*. 3rd ed. Bingley: Emerald Group Publishing, 2012.
- Dervin, Brenda, and Nilan, Michael. "Information Needs and Uses." *Annual Review of Information Science and Technology*, 21, 1986.
- Ingwersen, Peter. *Information Retrieval Interaction*. London: Taylor Graham, 1992.
- Kawatra, P.S. *Library User Studies: A Manual for Librarians and Information Scientists*. Bombay: Jaico Publishing House, 1992.
- Wilson, T.D. "Recent Trends in User Studies: Action Research and Qualitative Methods." *Information Research*, 5(3), 2000. http://informationr.net/ir/5-3/paper76.html

*** ***** ***

UNIT 10: STANDARDS AND PROTOCOLS IN IR

UNIT STRUCTURE

10.1 Learning Objectives
10.2 Introduction
10.3 Information Retrieval Standards and Protocols
  10.3.1 Z39.50
  10.3.2 How Z39.50 Works
  10.3.3 SRW (Search/Retrieve Web Service)
  10.3.4 SRU (Search/Retrieve via URL)
  10.3.5 CQL (Contextual Query Language)
10.4 Intelligent Information Retrieval
  10.4.1 Expert Systems
  10.4.2 Expert Systems for Library Professionals
  10.4.3 Expert Systems in Cataloguing
  10.4.4 Expert Systems in Classification
  10.4.5 Expert Systems in Document Delivery
  10.4.6 Expert Systems in Abstracting
10.5 Vocabulary Control in IR Systems
  10.5.1 Meaning and Need for Vocabulary Control
  10.5.2 Objectives and Requirements
10.6 Thesaurus
  10.6.1 Definition and History
  10.6.2 Structure and Content of a Thesaurus
  10.6.3 Types of Thesauri
  10.6.4 Relationships Between Terms
  10.6.5 Role of Thesaurus in Indexing and Searching
10.7 Let Us Sum up
10.8 Answer to Your Progress
10.9 Model Questions
10.10 References and Further Reading

10.1 Learning Objectives

After going through this unit, you will be able to:

- Describe the major standards and protocols used in information retrieval, including Z39.50, SRW, SRU, and CQL.

- Explain how Z39.50 operates within a client-server architecture.
- Define intelligent information retrieval and understand the role of expert systems in library and information science.
- Describe the applications of expert systems in cataloguing, classification, document delivery, and abstracting.
- Explain the concept of vocabulary control and the role of the thesaurus as a tool for information retrieval.

10.2 Introduction

The rapid development of Information and Communication Technology (ICT) has fundamentally transformed the manner in which information is searched for and retrieved. Modern information retrieval (IR) systems operate across distributed and heterogeneous networks, necessitating the adoption of shared standards and protocols that ensure interoperability. Alongside these technological frameworks, the application of artificial intelligence techniques such as expert systems has extended the intelligence and automation possible in library-based operations. At the foundation of any effective IR system lies the control of vocabulary, a discipline that ensures consistent representation and retrieval of information through tools such as the thesaurus.

This unit examines three core dimensions of advanced information retrieval: the standards and protocols that govern machine-to-machine communication, the intelligent systems that support decision-making in libraries, and the vocabulary control mechanisms that bridge the gap between natural language and formal indexing languages.

10.3 Information Retrieval Standards and Protocols

A standard may be understood as an agreement on the method by which a task is to be performed so as to yield a predictable result. According to the National Information Standards Organization (NISO, USA), all standards are established through an agreement based on the expertise of designers, application developers, vendors, and product users, and are subsequently approved by the American National Standards Institute (ANSI). Standards and protocols for IR systems ensure that different systems can communicate, share data, and provide interoperable search and retrieval services across diverse platforms.

10.3.1 Z39.50

Z39.50 is a non-proprietary and vendor-independent standard protocol that defines computer-to-computer information retrieval technique, used at both the national and international level. It was originally approved by NISO in 1988. In 1998, the

International Organization for Standardization (ISO) adopted Z39.50 and issued it as ISO 23950: "Information and documentation: Information retrieval (Z39.50)".

The primary goal of Z39.50 is to reduce the complexity and difficulties involved in searching and retrieving electronic information. Using Z39.50, a user operating through their own system can search and retrieve information from other Z39.50-compliant computer systems without having any prior knowledge of the search syntax used by those other systems. The protocol operates within a client-server architecture and acts as a common language that all Z39.50-enabled systems can understand. As Dutta describes it, "It is like the Esperanto language, which bridges several languages and dialects that various information systems speak".

Z39.50 standardises the messages used by clients and servers for communication, regardless of what software, platform, or hardware is employed. A Z39.50-enabled client system can communicate with diverse servers, and a Z39.50-enabled server is searchable by client systems developed by different vendors. TCP/IP (Transmission Control Protocol/Internet Protocol) is used as the standard transport mechanism by almost all Z39.50 implementations.

10.3.2 How Z39.50 Works

Z39.50 governs the entire process by which a client translates a query into a standard format to be sent to a server. The following describes its operational sequence:

1. A user formulates a query through the client-side Z39.50-enabled interface.
2. The Z39.50 protocol translates this query into a standard, platform-independent format and transmits it to the remote server.
3. Upon receiving the query, the server applies the Z39.50 rules to translate the query into a format that the local database understands.
4. The server performs the search and returns the results to the client.
5. The interface software at the client processes the returned results and displays them as closely as possible to the way records are displayed in the user's native system.

Since all associated technical activities occur behind the scenes, users only see their familiar search and display interface. Z39.50 thereby makes the wealth of information available on distributed systems accessible without requiring users to learn multiple search syntaxes.

10.3.3 SRW (Search/Retrieve Web Service)

SRW stands for Search/Retrieve Web Service protocol. Its aim is to minimise the cross-language problems that arise in distributed information retrieval. The goal is to allow access to several networked resources and support interoperability among distributed databases, using a common utilisation framework. SRW was developed by

collective implementers with more than twenty years of experience with the Z39.50 protocol, and builds upon the nascent developments in the technological arena of the web.

SRW provides both Simple Object Access Protocol (SOAP) and URL-based access mechanisms to a wide range of possible clients, from Microsoft's .Net initiative to simple Extensible Stylesheet Language Transformations (XSLT) and JavaScript transformations. It directs the usage of open and industry-supported standards like eXtensible Markup Language (XML), XML Schema, SOAP, and XPath. SRW defines a web service incorporating several Z39.50 features, most notably the Search, Present, and Sort Services. Because it has been developed on Z39.50 semantics, it makes it easier for existing Z39.50 systems to transition, decreasing the barriers to new information providers who wish to make their contents available via a standard search and retrieve mechanism.

SRW also provides semantics in search of databases having both metadata and objects, including text and non-text materials. It influences the use of the Contextual Query Language (CQL), discussed below, which provides expressive but intuitive ways for search formulation.

10.3.4 SRU (Search/Retrieve via URL)

SRU stands for Search/Retrieve via URL. It is a standard XML-based protocol for search, utilising CQL (available at http://www.loc.gov/cql), a standard syntax for query representation. The prime difference between SRU and SRW is the transport mechanism: SRU uses HTTP as the transport mechanism while SRW is based on the SOAP protocol and uses XML streams for both the query and the results. In SRU, the query is communicated as a URL and the XML result is received as if it were a web page. The POST method, an alternate means of using HTTP transport, is not acceptable within the SRU protocol.

The advantage of SRU is that a wide variety of transport mechanisms can be used, including, for instance, e-mail. This makes SRU particularly flexible in contexts where SOAP-based communication may not be practical.

10.3.5 CQL (Contextual Query Language)

CQL stands for Contextual Query Language, formerly known as Common Query Language (http://www.loc.gov/cql). It is designed for use with SRW, which is a search protocol successor to Z39.50. CQL is an abstract and extensible query language created for maximum interoperability among connected systems. The goal is to reduce the difficulty of learning and using a query language while retaining the capability to allow complex searches.

CQL provides a standards-tested mechanism to specify a query to select records from a database that may be used either internally or remotely. Although it is primarily used in the bibliographic domain, it is not restricted to this context alone. Notably, the

bibliographic database of the Library of Congress has an SRW/CQL interface available for all 28 million records. CQL can also be used in applications such as Open Office to identify, locate, and integrate large amounts of data within the application environment.

Check Your Progress - I

Q1. What does Z39.50 stand for, and which organisation first approved it?

Q2. Fill in the blank: SRW stands for Web Service protocol.

Q3. What is the key difference between SRW and SRU?

Q4. CQL was formerly known as .

Q5. Match the following:

Column A	Column B
(a) Z39.50	(i) Uses HTTP as transport
(b) SRW	(ii) ISO 23950
(c) SRU	(iii) Uses SOAP protocol
(d) CQL	(iv) Contextual Query Language

10.4 INTELLIGENT INFORMATION RETRIEVAL

"According to Sparck Jones in Intelligent information retrieval (1983), intelligent information retrieval is a computer system having the capability to infer knowledge with the help of its previous knowledge for establishing a link between the requirement of its user and a set of candidate documents". This definition underscores the fundamental aspiration of intelligent IR: to move beyond mere keyword matching towards systems that can reason, infer, and adapt. The realisation among researchers that knowledge must be incorporated into IR systems led them to consider artificial intelligent approaches, among which the expert system emerged as a particularly significant class.

10.4.1 Expert Systems

"According to Peter Jackson in Introduction to Expert Systems (1998), an expert system is a computer system which emulates the decision-making ability of human experts". Expert systems are designed to solve complex problems by reasoning over knowledge stored in a knowledge base. The knowledge in the knowledge base is primarily represented as IF-THEN rules rather than conventional procedural code.

An expert system has two primary components: the inference engine and the knowledge base. A knowledge base consists of rules and facts, representing the knowledge about real world objects. The inference engine applies these rules to the known facts for deducing new facts or knowledge. Inference engines can also provide

explanations for the deduced knowledge. The knowledge bases are structured in a form of classes, subclasses, and instances, drawing on techniques similar to object-oriented programming.

The first expert systems were invented in the 1970s and proliferated in the 1980s. Among the most important inference techniques adopted into expert systems, as identified by Mettrey (1987), are the following:

- **Truth Maintenance:** These systems record the dependencies in a knowledge base so that when facts are changed, the dependent knowledge is also altered accordingly.
- **Hypothetical Reasoning:** The knowledge base is subdivided into several possible views or "worlds," enabling the inference engine to explore several possibilities in parallel.
- **Fuzzy Logic:** Each rule is associated with a probability value, enabling assertions to be made with degrees of certainty rather than absolute truth or falsity.
- **Ontology Classification:** With the introduction of object classes in knowledge bases, it becomes possible to reason about the structure of objects, not merely their values. Such inference engines are known as classifiers and are considered a key technology for the Semantic Web.

10.4.2 Expert Systems for Library Professionals

Expert systems can be employed to emulate the tasks of library professionals, particularly in areas that involve intelligent activity normally performed by domain experts. As Schoech (1999) notes, expert systems are designed to make decisions, and not just to produce reports. Their specificity of goal distinguishes them from general information systems. Expert systems have been developed in several core areas of library and information science (LIS), including cataloguing, classification, document delivery, and abstracting, each of which is examined in the following sections.

10.4.3 Expert Systems in Cataloguing

Expert systems in cataloguing have been developed to automate or semi-automate the generation of bibliographic records. One significant early system was AUTOCAT, produced in Germany, designed to generate bibliographic records of physical sciences periodicals available in machine-readable form (Endres-Niggemeyer and Knorz, 1987).

Another notable project was conducted at OCLC, where Weibel, Oskins, and Vizine-Goetz built a prototype known as the OCLC Automated Title Page Cataloguing Project, a rule-based tool designed to prepare descriptive cataloguing from title pages (Weibel et al., 1989). A further significant project, Qualcat (Quality Control in

Cataloguing), was undertaken at the University of Bradford. The goals of Qualcat were to develop expert systems to select the best records, to link databases with centralised authority control, to build a fully automated control package for day-to-day running, and to investigate interface problems for cataloguing (British Library Research and Development, 1994).

10.4.4 Expert Systems in Classification

Classification is a particularly challenging task for expert systems, primarily because the relationships between documents and classes are often not explicit, even when the classification schedules are available. Several expert systems have nevertheless been developed for item, patent, and book classification, by researchers including Sharif (1988), Cosgrove and Weimann (1992), Valkonen and Nykanen (1991), and Gopinath and Prasad (1994).

In 1986, Paul Burton conducted an exploratory research project at the University of Strathclyde, UK, to assess the merits of different ways of knowledge representation and the suitability of expert systems for classification. The outcome was a prototype expert system capable of providing a Dewey class number based on the information provided by the user. In a related development, OCLC developed an expert system called Cataloguer's Assistant, tested at Carnegie-Mellon University to reclassify a mathematics and computer science collection.

10.4.5 Expert Systems in Document Delivery

There are relatively few reported applications of expert systems specifically in document delivery. Brown (1993) explained the use of expert system technologies in the equipment division of Raytheon Company for coordinating requests for specifications and standards documents with purchases made through the acquisitions unit. Abate (1995) reported on an expert system developed for the library of a law firm, designed for document delivery in decision-making, using the expert system shell VP-Expert.

Although these applications are limited in number, they demonstrate the feasibility of automating coordination processes in document supply chains within specialised library environments.

10.4.6 Expert Systems in Abstracting

The application of expert systems in abstracting has been primarily focused on the automatic generation of abstracts from scientific journal articles and conference proceedings. H. P. Luhn's work in 1958 is recognised as the first reported work on automatic abstracting.

DeJong (1982) developed a system known as FRUMP, which analysed articles from newspapers using frame-based techniques. Articles were first scanned, and data

were automatically fed into different slots within frames. Scripts were then executed to generate summaries of the information held in the relevant frames. Rau, Jacobs, and Zernik (1989) developed a system known as SCISOR that generated reports on corporate acquisitions and mergers, extending abstracting research beyond scientific articles to other categories of textual material.

It is worth noting that a very few expert systems are reportedly in full operational use across the various areas of LIS. Some systems progressed to commercial deployment but were later withdrawn, such as Tome Searcher and Tome Selector. The field nevertheless continues to offer productive avenues for research and development in automated library services.

Check Your Progress - II

Q1. Name the two primary components of an expert system.

Q2. Fill in the blank: The OCLC expert system for classification was called .

Q3. Which system was developed by DeJong for automatic abstracting using frame-based techniques?

Q4. Match the following:

Column A	Column B
(a) AUTOCAT	(i) Document delivery, law firm
(b) Qualcat	(ii) Bibliographic records, Germany
(c) Abate (1995)	(iii) Quality control in cataloguing
(d) SCISOR	(iv) Corporate acquisitions abstracting

Q5. True or False: Expert systems are designed to produce reports, not to make decisions.

10.5 Vocabulary Control in IR Systems

10.5.1 Meaning and Need for Vocabulary Control

In any IR system, documents are indexed using terms drawn from a vocabulary. When that vocabulary is uncontrolled, indexers and searchers may independently use different terms to represent the same concept, leading to failures in retrieval. "The term vocabulary control refers to a limited set of terms that must be used to index documents, and to search for these documents, in a particular system. It may be defined as a list of terms showing their relationships and used to represent the specific subject of a document".

The need for vocabulary control arises because natural language contains a large number of synonyms, quasi-synonyms, homonyms, acronyms, and ambiguous terms. If vocabulary control is not exercised, different indexers or the same indexer may use different terms for the same concept on different occasions during indexing. Similarly, a user's search term may not match the term assigned by an indexer,

resulting in retrieval failure. This problem is sometimes referred to as the "vocabulary problem" in human-system communication (Furnas et al., 1987).

A controlled vocabulary is a limited set of terms, selected from natural language, which must be used consistently by indexers and searchers. A certain degree of structure is introduced into the controlled vocabulary so that terms whose meanings are related are brought together or linked in a defined way.

10.5.2 Objectives and Requirements

The two fundamental objectives of vocabulary control are:

1. To ensure the consistent representation of the subject matter of documents at the time of indexing and searching, achieved through the control of synonymous and near-synonymous expressions and the disambiguation of homographs.
2. To facilitate the conduct of comprehensive searches by linking together terms that are closely related semantically.

The requirements of a controlled vocabulary include: adequate literary warrant (terms justified by the literature and information needs of users); sufficient specificity to allow precise searches; promotion of consistency through control of synonyms and near-synonyms; and reduction of terminological ambiguity through scope notes and qualifiers.

10.6 THESAURUS

10.6.1 Definition and History

The word "thesaurus" derives from the Greek term "thesauros," meaning a storehouse or treasury of words. The Oxford English Dictionary traces its usage in the sense of "a treasury of knowledge" to 1736. The modern concept of the thesaurus as a classification of ideas is associated with Peter Mark Roget, whose literary thesaurus was first published in 1852. However, Roget's work had no direct connection with information retrieval.

Helen Brownson is credited with first using the term "thesaurus" in the context of information retrieval, at the Dorking Conference on Classification Research in 1957. H. P. Luhn was among the first to conceptualise an IR thesaurus, suggesting the compilation of "families of notions" and a "dictionary of notion families" for indexing purposes. The first thesaurus deployed in an information retrieval system was developed by Du Pont in the USA around 1959.

In terms of its formal definition, the International Organization for Standardization (ISO) provides two complementary characterisations. In terms of function: "According to ISO 25964-1 (International Organization for Standardization, 2011), a thesaurus is a controlled and structured vocabulary in which concepts are represented by terms,

organized so that relationships between concepts are made explicit, and preferred terms are accompanied by lead-in entries for synonyms or quasi-synonyms". In terms of structure, the ISO standard states that a thesaurus is "a controlled and dynamic vocabulary of semantically and generically related terms which covers a specific domain of knowledge".

The American standard ANSI/NISO Z39.19-2005 offers a broadly compatible definition: "A controlled vocabulary arranged in a known order and structured so that the various relationships among terms are displayed clearly and identified by standardized relationship indicators. Relationship indicators should be employed reciprocally" (National Information Standards Organization, 2005/2010).

10.6.2 Structure and Content of a Thesaurus

The essential core of a thesaurus is a collection of concepts represented by terms and interlinked by relationships. The three main types of relationships are equivalence (between terms), hierarchical (between concepts), and associative (between concepts). By long-established convention, the tags USE and UF (Use For) precede preferred and non-preferred terms respectively, while the tags BT (Broader Term), NT (Narrower Term), and RT (Related Term) indicate hierarchical and associative relationships.

Each entry in a thesaurus typically consists of the following components:

text

DESCRIPTOR

[Scope note, if needed]

USE / UF [Synonyms and quasi-synonyms – Equivalence relationship]

BT [Broader Terms – Hierarchical, superordinate]

NT [Narrower Terms – Hierarchical, subordinate]

RT [Related Terms – Associative relationship]

TT [Top Term – Highest superordinate in hierarchy]

A thesaurus may be arranged alphabetically, by classified hierarchy, or with a graphical display. An alphabetical thesaurus lists descriptors with their relationships in alphabetical order, while a classified thesaurus lists descriptors in accordance with the hierarchical relationships, with appropriate indentation showing levels of hierarchy.

10.6.3 Types of Thesauri

Based on the nature of terminology control, thesauri are classified into two main types:

- **Controlled thesauri:** Allow only one term (the preferred term or descriptor) to denote a concept for the purposes of indexing and searching. These can be maintained manually.
- **Free language thesauri:** Allow use of all terms denoting a concept to be used for indexing and searching. These require machine maintenance and retrieval.

Beyond this basic classification, thesauri may also be distinguished by language coverage (monolingual or multilingual), scope (macro-thesauri covering broad subject areas versus micro-thesauri with high specificity), and purpose (indexing thesauri, search thesauri, or integrated tools such as the Thesauro-facet and Classaurus).

Thesauro-facet was developed by Jean Aitchison and others. It is fundamentally a faceted classification integrated with a thesaurus, comprising a faceted classification scheme and an alphabetical thesaurus. The link between the two is provided by the class notation. It can be used in both pre- and post-coordinate indexing systems.

Classaurus was developed by Dr. Ganesh Bhattacharyya at DRTC (Documentation Research and Training Centre). It incorporates features of both a faceted classification scheme and a conventional alphabetical thesaurus. It is an elementary category-based faceted systematic scheme of hierarchical classification in the verbal plane. Unlike a standard thesaurus, a Classaurus does not include associatively related terms (RT), as its category-based faceted structure renders such enumeration unnecessary.

10.6.4 Relationships Between Terms

The inter-relationships between terms in a thesaurus are expressed through two broad categories of relationships:

1. Hierarchical Relationships

These express superordinate and subordinate relationships. There are three types:

- **Genus-Species (Generic) Relationship:** Links genus and species; e.g., Rodents (BT) - Mice (NT).
- **Hierarchical Whole-Part Relationship:** The name of a part implies the name of its whole in any context; e.g., Cardio-vascular System (NT) - Circulatory System (BT).
- **Instance Relationship:** Links a proper name with a common noun; e.g., Alps (NT) - Mountain Regions (BT).

2. Non-Hierarchical Relationships

- **Equivalence (Preferential) Relationship:** Refers to preferred terms and distinguishes them from non-preferred terms. Synonyms, near-synonyms, and quasi-synonyms are grouped under this category, represented by the symbols USE and UF.
- **Associative (Affinitive) Relationship:** Covers other relationships between terms that are related but are neither consistently hierarchical nor equivalent. Examples include a discipline and its object of study (e.g., Birds RT Ornithology), an action and its property (e.g., Indexing RT Efficiency), and an action and its resulting product (e.g., Weaving RT Cloth). These are represented by the code RT (Related Term).

10.6.5 Role of Thesaurus in Indexing and Searching

The thesaurus contributes to consistency in indexing between different indexers and reduces incorrect term assignments, particularly where descriptors are carefully distinguished through scope notes. It plays two principal roles:

- **Prescriptive role:** It prescribes the term to be assigned. The USE reference is a prescriptive indicator that directs the indexer or searcher from a non-preferred term to the authorised descriptor.
- **Suggestive role:** It suggests terms to be considered in addition to or instead of a given term. The RT, and to a certain extent the BT and NT, function as suggestive indicators.

"According to Dextre Clarke in The Information Retrieval Thesaurus (2019), the traditional aim of a thesaurus is to guide the indexer and the searcher to choose the same term for the same concept". In the context of computer-based searching, the electronic thesaurus further enables functions such as search explosion (expanding a search via hierarchical relationships to retrieve all narrower concept terms automatically) and search expansion (extending a search via associative relationships). The thesaurus also provides a map of a given subject field, which assists users in understanding the structure of that field. In post-coordinate indexing systems, the thesaurus aids the searcher in choosing appropriate terms and provides alternative search paths, especially when searching an unfamiliar domain.

Check Your Progress - III

Q1. Who first used the term "thesaurus" in the context of information retrieval?

Q2. Fill in the blank: The symbols BT and NT in a thesaurus stand for and respectively.

Q3. What is the difference between a Thesauro-facet and a Classaurus?

Q4. Match the following:

Column A	Column B
(a) USE/UF	(i) Associative relationship
(b) BT/NT	(ii) Equivalence relationship
(c) RT	(iii) Hierarchical relationship
(d) TT	(iv) Top Term

Q5. True or False: A Classaurus includes Related Terms (RT) just as a standard thesaurus does.

10.7 LET US SUM UP

This unit has covered three core aspects of advanced information retrieval. In the domain of standards and protocols, Z39.50 was examined as a foundational client-server standard for cross-system bibliographic searching, alongside its successors SRW and SRU, and the query language CQL that underpins them. In the domain of intelligent IR, expert systems were discussed as computer systems that emulate expert decision-making through inference engines and knowledge bases. Their applications in cataloguing (AUTOCAT, OCLC Automated Title Page Cataloguing Project), classification (Cataloguer's Assistant), document delivery (Brown, 1993; Abate, 1995), and abstracting (FRUMP, SCISOR) illustrate the range of library operations that can be supported by intelligent systems. Finally, vocabulary control and the thesaurus were examined as the terminological infrastructure of IR: the thesaurus provides preferred terms, controls synonymy, represents hierarchical and associative relationships, and guides both indexers and searchers towards consistent and effective retrieval.

10.8 ANSWER TO CHECK YOUR PROGRESS

Check Your Progress 1

Ans to Q1. Z39.50 is an IR standard protocol defining computer-to-computer information retrieval. It was originally approved by the National Information Standards Organization (NISO), USA, in 1988.

Ans to Q2. Search/Retrieve.

Ans to Q3. SRW uses SOAP protocol and XML streams for both query and results, while SRU uses HTTP as the transport mechanism and communicates the query as a URL.

Ans to Q4. Common Query Language.

Ans to Q5. (a) - (ii); (b) - (iii); (c) - (i); (d) - (iv).

Check Your Progress 2

Ans to Q1. The two primary components of an expert system are the inference engine and the knowledge base.

Ans to Q2. Cataloguer's Assistant.

Ans to Q3. FRUMP.

Ans to Q4. (a) - (ii); (b) - (iii); (c) - (i); (d) - (iv).

Ans to Q5. False. Expert systems are designed specifically to make decisions, and not merely to produce reports (Schoech, 1999).

Check Your Progress 2

Ans to Q1. Helen Brownson is credited with first using the term "thesaurus" in the context of information retrieval, at the Dorking Conference on Classification Research in 1957.

Ans to Q2. Broader Term; Narrower Term.

Ans to Q3. A Thesauro-facet (Aitchison) integrates a faceted classification with an alphabetical thesaurus, usable in both pre- and post-coordinate systems. A Classaurus (Bhattacharyya) also combines faceted classification with a thesaurus, but does not enumerate Related Terms (RT), relying instead on the document's subject content to determine associative relationships.

Ans to Q4. (a) - (ii); (b) - (iii); (c) - (i); (d) - (iv).

Ans to Q5. False. A Classaurus does not include associatively related terms (RT), unlike a standard thesaurus.

10.9 Model Questions

Short Questions

1. What is Z39.50? State its primary goal.
2. Distinguish between SRW and SRU.
3. What is CQL, and in which domain is it primarily used?
4. Define an expert system with reference to an authority.
5. Name any two expert systems developed for library cataloguing.
6. What is the vocabulary problem in information retrieval?
7. Give the ISO definition of a thesaurus.
8. What are the main types of relationships displayed in a thesaurus?
9. Distinguish between a Thesauro-facet and a Classaurus.
10. What are the prescriptive and suggestive roles of a thesaurus?

Long Questions

1. Describe the Z39.50 protocol in detail, including its architecture, operational mechanism, and significance for distributed information retrieval.
2. Write a comprehensive account of the applications of expert systems in library and information services, with reference to specific systems developed for cataloguing, classification, document delivery, and abstracting.
3. Explain the concept of vocabulary control in information retrieval systems. Discuss its objectives, requirements, and the role of controlled vocabularies in ensuring retrieval effectiveness.
4. What is a thesaurus? Discuss its structure, types of relationships, and its role in indexing and searching within an information retrieval system.
5. Compare and contrast SRW, SRU, and CQL as protocols and query languages in the context of modern information retrieval. How do they relate to the Z39.50 standard?

10.10 REFERENCES AND FURTHER READING

References

- Abate, A. K. (1995). Document delivery expert. *Journal of Interlibrary Loan, Document Delivery and Information Supply*, 6(1), 17-37.
- Aitchison, J. and Gilchrist, A. (1987). *Thesaurus Construction: A Practical Manual*. 2nd ed. London: ASLIB.
- Bhattacharyya, G. (1982). Classaurus: Its fundamental, design and use. Paper presented at the Fourth International Study Conference on Classification Research, Augsburg, June 28-July 2, 1982.
- British Library Research and Development. (1994). *Qualcat: Automation of Quality Control in Cataloguing*, pp. 1-12.
- Brown, Lynne C. Branche. (1993). Expert systems and document delivery in an automated acquisitions environment. *Library Acquisitions: Practice and Theory*, 17(3), 353-357.
- Cosgrove, S. J. and Weimann, J. M. (1992). Expert system technology applied to item classification. *Library Hi Tech*, 10(1/2), 33-40.
- DeJong, Gerald. (1982). An overview of the FRUMP system. In Lehnert, W. G. and Ringle, M. H. (eds.), *Strategies for Natural Language Processing*. London: Lawrence Erlbaum, pp. 149-172.
- Dextre Clarke, Stella G. (2019). The information retrieval thesaurus. *Knowledge Organization*, 46(6), 439-459. DOI: 10.5771/0943-7444-2019-6-439.
- Dutta, Biswanath. (n.d.). Evolutions in Information Retrieval. [Chapter 12]. New Delhi: CEC-UGC.
- Endres-Niggemeyer, B. and Knorz, G. (1987). AUTOCAT: Knowledge-based descriptive cataloguing of articles published in scientific journals. *Knowledge Based Systems*, Munich, October 20-21, 1987.
- Furnas, G. W., Landauer, T. K., Gomez, L. M., and Dumais, S. T. (1987). The vocabulary problem in human-system communication. *Communications of the ACM*, 30(11), 964-971.
- Gopinath, M. A. and Prasad, A. R. D. (1994). A knowledge representation model for analytico-synthetic classification. In *Knowledge Organization and Quality Management: Proceedings of the 3rd ISKO Conference*, Copenhagen, June 20-24, 1994, pp. 320-327.
- ISO (International Organization for Standardization). (2011). *Thesauri for Information Retrieval. Part 1 of Thesauri and Interoperability with Other Vocabularies*. ISO 25964-1. Geneva: ISO.
- ISO (International Organization for Standardization). (2013). *Interoperability with Other Vocabularies. Part 2 of Thesauri and Interoperability with Other Vocabularies*. ISO 25964-2. Geneva: ISO.
- Jackson, Peter. (1998). *Introduction to Expert Systems*. 3rd ed. Addison-Wesley. ISBN 978-0-201-87686-4.

- Lancaster, F. W. (1986). *Vocabulary Control for Information Retrieval*. 2nd ed. Arlington, VA: Information Resources Press.
- Mettrey, William. (1987). An assessment of tools for building large knowledge-based systems. *AI Magazine*, 8(4).
- National Information Standards Organization (NISO). (2005/2010). *Guidelines for the Construction, Format, and Management of Monolingual Controlled Vocabularies*. ANSI/NISO Z39.19-2005 (R2010). Bethesda, MD: NISO.
- Rau, L. F., Jacobs, P. S. and Zernik, U. (1989). SCISOR: Information extraction and text summarization using linguistic knowledge acquisition. *Information Processing and Management*, 25(4), 419-428.
- Schoech, D. (1999). *Human Services Technology: Understanding, Designing and Implementing Computer and Internet Applications in the Social Services*. 2nd ed. New York: Haworth.
- Sharif, Carolyn A. Y. (1988). *Developing an Expert System for Classification of Books Using Micro-Based Expert Systems Shells*. British Library Research Paper 32.
- Silva, Sharon Manel De. (1997). A review of expert systems in library and information science. *Malaysian Journal of Library and Information Science*, 2(2), 57-92.
- Sparck Jones, K. (1983). Intelligent retrieval. In Jones, K. P. (ed.), *Intelligent Information Retrieval: Informatics 7*. London: Aslib, pp. 136-142.
- Valkonen, P. and Nykanen, O. (1991). An expert system for patent classification. *World Patent Information*, 13(3), 143-148.

Further Reading

- Aitchison, Jean, Alan Gilchrist and David Bawden. (2000). *Thesaurus Construction and Use: A Practical Manual*. 4th ed. London: Aslib.
- Broughton, Vanda. (2006). *Essential Thesaurus Construction*. London: Facet.
- Chowdhury, G. G. (1999). *Introduction to Modern Information Retrieval*. London: Library Association Publishing.
- Foskett, A. C. (1996). *The Subject Approach to Information*. 5th ed. London: Library Association Publishing.
- Morris, Anne (ed.). (1992). *The Application of Expert Systems in Libraries and Information Centres*. London: Bowker-Saur.
- NISO. Z39.50: A Primer on the Protocol. Available at: www.niso.org/publications/press/Z3950primer.pdf
- Shiri, Ali. (2012). *Powering Search: The Role of Thesauri in New Information Environments*. Medford, NJ: Information Today.

*** ***** ***

UNIT 11: STATISTICAL METHODS FOR INFORMATION RETRIEVAL

UNIT STRUCTURE

11.1 Learning Objectives
11.2 Introduction
11.3 Statistical Methods in Information Retrieval
  11.3.1 The Role of Statistics in IR
  11.3.2 Key NLP Pre-Processing Techniques
  11.3.3 Drawbacks of the Boolean Model
11.4 The Boolean Model
  11.4.1 Definition and Structure
  11.4.2 Extensions to the Basic Boolean Model
  11.4.3 Drawbacks of the Boolean Model
11.5 The Vector Space Model (VSM)
  11.5.1 Conceptual Foundation
  11.5.2 Term Weighting: TF-IDF
  11.5.3 Advantages and Limitations of the VSM
  11.5.4 Relevance Feedback in the VSM
11.6 The Probabilistic Model
11.7 Comparative Analysis of IR Models
11.8 Practical Implementations and Evaluation
11.9 Let Us Sum Up
11.10 Answer to Check Your Progress
11.11 Model Questions
11.12 References and Further Readings

11.1 Learning Objectives

After going through this unit, you will be able to:

- Describe the role of statistical methods and standard pre-processing techniques in information retrieval systems.
- Explain the Boolean retrieval model and evaluate its limitations for ranked document retrieval.
- Illustrate how the Vector Space Model uses TF-IDF weighting and cosine similarity to rank documents.
- Apply the statistical language modeling framework to formally account for stop words, mandatory terms, and morphological variation.

11.2 Introduction

Information retrieval (IR) is fundamentally concerned with the problem of locating relevant documents from a large collection in response to a user's query. According to Hull in *Information Retrieval Using Statistical Classification*, "an information retrieval system informs the user of the existence and whereabouts of documents relating to his request. It does not inform the user on the subject of his request." This distinction is significant: IR systems do not interpret meaning in the human sense; they rely on statistical and structural models to estimate relevance. The present unit examines the formal statistical models that underpin modern IRspecifically, the Boolean model, the Vector Space Model (VSM), and the probabilistic model and traces the evolution from rule-based retrieval to statistically grounded, ranked retrieval.

11.3 Statistical Methods in Information Retrieval

11.3.1 The Role of Statistics in IR

Full-text information retrieval requires mechanisms that go far beyond simple keyword lookup. As Hiemstra and de Jong (2001) observe, "information retrieval systems that only use word statistics treat words like 'milk', 'cow', 'cows', 'cattle', etc., as if they are totally unrelated," which makes it difficult to incorporate even basic linguistic relationships into formal retrieval models. Statistics provide the mathematical foundation to move beyond this limitation by modeling word frequency distributions, co-occurrence patterns, and probabilistic relevance estimates.

The primary challenge is that a query submitted to any large document collection typically matches thousands of documents, but only a small number are genuinely relevant to the user's need. A user of a web search engine will ordinarilyexamine only the first ten to twenty results returned, making the ranking of documents not merely their retrieval, the central problem. Statistical methods address this by assigning numerical scores to documents reflecting their estimated relevance, enabling ranked output rather than unordered result sets.

11.3.2 Key NLP Pre-Processing Techniques

Before statistical models can be applied, documents and queries undergo pre-processing. The standard pipeline includes tokenization, stop-word removal, and morphological normalization (stemming or lemmatization). According to Hiemstra and de Jong (2001), "a typical approach to document indexing and query processing is the following: first a tokenization process takes place, then stop words are removed, and finally the remaining words are stemmed."

Tokenization converts raw text into a sequence of processing tokens by identifying word symbols and inter-word symbols, discarding punctuation and other non-alphabetic characters.

Stop-word removal eliminates high-frequency words (such as "the," "it," and "a") that carry little semantic information. Removing 200 to 300 of the highest-frequency words in a collection can reduce index size by 30 to 50 percent. Stop-word lists may also include infrequent function words that are semantically empty.

Stemming applies morphological rules to reduce words to a common root form. The Porter stemmer (1980) is the most widely accepted algorithm. However, stemming "tends to help as many queries as it hurts" and may conflate semantically distinct words, causing users to question the system's integrity. Dictionary-based lemmatization achieves linguistically correct output but requires part-of-speech disambiguation.

Phrase extraction treats multi-word units as single tokens. There are three approaches: predefined phrases, statistically co-occurring non-stop-word pairs, and syntactically parsed noun phrases. Evaluations show that phrase extraction, like stemming, does not improve retrieval effectiveness significantly on its own.

Check Your Progress – I

Q1. What is the primary purpose of stop-word removal in IR pre-processing?

Q2. Match the following:

Column A	Column B
Porter Algorithm	Multi-word processing token
Tokenization	Morphological normalization
Phrase extraction	Conversion of text to word sequences

Q3. Fill in the blank: Removing 200–300 highest-frequency words from the index can reduce index size by to percent.

Q4. True or False: Stemming always improves retrieval effectiveness.

11.4THE BOOLEAN MODEL

11.4.1 Definition and Structure

According to Kowalski and Maybury (2002) in *Information Storage and Retrieval Systems*, "Boolean logic allows a user to logically relate multiple concepts together to define what information is needed." The Boolean retrieval model is the oldest and most straightforward formal model of IR. It represents queries as logical expressions using the connectives AND (set intersection), OR (set union), and NOT (set complement),

which are applied to the index terms of documents. A document is retrieved if and only if it satisfies the logical expression of the query.

For example, the query COMPUTER OR PROCESSOR NOT MAINFRAME selects all documents discussing computers and/or processors that do not discuss mainframes. The Boolean model is computationally efficient: it uses an inverted index to look up each query term and merges the resulting document lists according to the Boolean constraints.

11.4.2 Extensions to the Basic Boolean Model

Modern Boolean systems extend the basic AND/OR/NOT operators with additional capabilities. The ADJ (adjacency) operator matches documents that contain two specified terms in direct sequence; for example, milk ADJ cow retrieves documents containing the phrase "milk cow." The NEAR operator allows matches within a specified word-distance window. These proximity-based extensions increase precision by constraining term co-occurrence to meaningful syntactic contexts.

The special variant known as M-of-N logic allows a user to specify that any item containing a minimum subset of listed terms is acceptable, which can be expanded into a combination of AND and OR operators.

11.4.3 Drawbacks of the Boolean Model

The fundamental limitations of Boolean retrieval can be summarized as:

- No control over the size of the output
- Not robust: the query is extremely sensitive to the choice of terms
- No concept of term importance
- No ranking of document importance
- Difficult to formulate and modify complex queries

The union of frequently occurring terms can retrieve far more documents than a user can examine, while the intersection of rare terms may retrieve nothing. There is also no automatic mechanism by which a found relevant document can be used to refine the query. The Boolean model "does not use word statistics and does not rank the documents." These shortcomings motivated the development of ranked retrieval models.

Check Your Progress – II

Q1. Which Boolean operator corresponds to set intersection?

Q2. Fill in the blank: The operator in Boolean retrieval matches documents containing two terms within a specified window of words.

Q3. List any two drawbacks of the Boolean model.

Q4. Match the following:

Column A	Column B
AND	Set union
OR	Set complement
NOT	Set intersection

11.5 The Vector Space Model (VSM)

11.5.1 Conceptual Foundation

The Vector Space Model (VSM), developed primarily by Salton and colleagues, represents both documents and queries as vectors in a high-dimensional term space. According to Hull (n.d.), "the Vector Space Model (VSM) represents queries and documents as vectors of term weights. Each element of the document vector measures the importance of a specific term in the document." If $Q_i = (q_{i1}, q_{i2}, \dots, q_{in})$is the query vector and $D_j = (d_{j1}, d_{j2}, \dots, d_{jn})$is the document vector, the similarity between the two is computed as their inner product:

$$\text{Sim}(Q_i, D_j) = \sum_k q_{ik} \cdot d_{jk}\text{Type equation here.}$$

Documents are then ranked in decreasing order of this similarity score and the highest-ranking documents are returned to the user.

11.5.2 Term Weighting: TF-IDF

Raw term frequency alone is an inadequate measure of term importance. A term that occurs frequently in a document may be common throughout the entire collection and therefore carry little discriminative information. The dominant term-weighting strategy combines two components:

Term Frequency (TF): The frequency of a term within a specific document, typically transformed logarithmically or by square root to prevent linear over-weighting.

Inverse Document Frequency (IDF): A measure of how rare a term is across the collection. According to Hull (n.d.), the IDF is expressed as:

$$\text{IDF}_i = \log \frac{N}{n_i}\text{Type equation here.}$$

where $N$is the total number of documents and $n_i$is the number of documents containing term i. A term that appears in only a few documents receives a high IDF weight and thus contributes strongly to document ranking when matched.

The combined TF-IDF weight for term $i$in document $j$is:

$$w_{ij} = f(d_{ij}) \cdot \log \frac{N}{n_i} \cdot f'(\| D_j \|)$$

where $f$and f'represent the term frequency function and the document-length normalization respectively. Salton and Buckley (1988) reported experiments with as many as 1,800 different variations of TF-IDF weights, reflecting the complexity of optimizing these components.

11.5.3 Advantages and Limitations of the VSM

The VSM addresses several key limitations of the Boolean model. Documents are ranked on a continuous scale, making it easy to control output size. Term weights capture relative importance, and the model accommodates both short and long queries as free text. Relevance feedback, the iterative improvement of a query using information from relevant documents already found, is elegantly supported because additional document text can be incorporated directly into the query vector.

However, the VSM assumes term independence: all terms are treated as orthogonal dimensions in the vector space. It is self-evident that terms have strong dependence patternsterms used in Chemistry are more likely to co-occur with each other than with vocabulary from Psychology. This independence assumption limits the model's ability to capture phrase meaning and semantic relationships. Additionally, longer documents tend to receive higher similarity scores simply because they contain more terms; document-length normalization is required to correct this bias.

11.5.4 Relevance Feedback in the VSM

One of the most powerful features of the VSM is its support for automatic relevance feedback. After an initial search, the user examines the highest-ranking documents and identifies those that are relevant. The system then constructs a revised query using the Rocchio method:

$$Q_{rfb} = kQ + \sum_{i \in \text{rel}} \frac{D_i}{| D_i |} - d \sum_{j \in \text{non-rel}} \frac{D_j}{| D_j |} \text{Type equation here.}$$

This strategy adds a weighted sum of relevant document vectors and subtracts a weighted sum of non-relevant document vectors from the original query. Relevance feedback using this approach yields substantial improvements in retrieval performance, ranging from 50 percent or more depending on the collection.

Check Your Progress – III

Q1. What does IDF stand for, and what does it measure?

Q2. Fill in the blank: In the VSM, documents and queries are represented as of term weights.

Q3. True or False: The VSM ranks documents on a continuous scale.

Q4. Which method is commonly used for automatic relevance feedback in the VSM? (a) Porter Algorithm (b) Rocchio Method (c) Hidden Markov Model (d) BM25

11.6THE PROBABILISTIC MODEL

11.6.1 Foundations of Probabilistic Retrieval

The probabilistic model of IR is grounded in the formal framework of probability theory rather than geometric similarity. Its central premise is that documents can be ranked according to the probability that they are relevant to a given query. According to Hull (n.d.), "the value of a document is measured by its relevance to the query, which is often expressed in the framework of a probabilistic model."

In the basic probabilistic framework, let $x = (x_1, x_2, \dots, x_n)$be a binary vector representing the presence or absence of each index term in a document. Using Bayes' Theorem, the decision rule for relevance becomes:

$$\frac{P(x \mid \text{rel}) \cdot P(\text{rel})}{P(x \mid \text{non-rel}) \cdot P(\text{non-rel})} > 1 \Rightarrow \text{relevant}$$

Assuming conditional independence between terms, the log-likelihood score can be expressed as a weighted sum involving estimated probabilities $p_i = P(x_i = 1 \mid \text{rel})$and $q_i = P(x_i = 1 \mid \text{non-rel})$for each term. These probabilities are estimated from relevance feedback data using a contingency table of term occurrences across relevant and non-relevant documents.

11.6.2 Statistical Language Models for IR

A more recent and theoretically unified probabilistic approach is the statistical language modeling (SLM) framework introduced in the context of IR by Hiemstra (1998), Miller et al. (1999), and Ponte and Croft (1998). According to Hiemstra and de Jong (2001), "a statistical language model assigns a probability to sequences of words. It does not distinguish between wellformed and unwellformed sequences: any sequence of words is considered, but some sequences are much more probable than others."

In this framework, a separate language model is estimated for each document in the collection. The model captures the probability that the document's language would "generate" the user's query. For information retrieval, this results in a document being ranked highly if the query terms have a high probability under its language model.

11.6.3 The Basic Mixture Model

Hiemstra and de Jong (2001) propose a mixture model that combines two probability estimates: the probability of a term in general language, $P(T)$, and the probability of a term in the relevant document's language, $P(T \mid D)$. The unknown parameter λ_i(where $0 \leq \lambda_i \leq 1$) represents the probability that a query term is "important" — that is, drawn specifically from the vocabulary of the relevant document rather than from general English. The basic model for a sequence of query terms $T_1, T_2, \ldots, T_n$is:

$$P(T_1, \ldots, T_n \mid D) = \prod_{i=1}^{n} \big((1-\lambda_i)P(T_i) + \lambda_i P(T_i \mid D)\big) \text{ Type equation here.}$$

Experimental studies show that the optimum value for $\lambda_i$lies between 0.15 and 0.3. This model outperforms ranked retrieval models that rely on the best-performing TF-IDF weighting algorithms, and it can be shown to have an equivalent TF-IDF-like weighting algorithm that produces the exact same ranked output.

11.6.4 Modeling Stop Words and Mandatory Terms

One significant advantage of the language modeling approach is its ability to formally justify common IR practices. Setting $\lambda_i = 0$for a term $T_i$means the term contributes equally to every document's score regardless of whether it appears in that document — it has no effect on the final ranking, mathematically formalizing stop-word removal. Setting $\lambda_i = 1$means that the probability of the query given the document is zero unless the term appears in the document — the term becomes mandatory.

As Hiemstra and de Jong (2001) observe, "to our knowledge, none of the other existing models of ranked retrieval can mathematically explain or justify the use of stop words and/or mandatory terms, but they play a role in many practical information retrieval systems."

11.6.5 Modeling Phrases, Stemming, and Translation

The language modeling framework also accommodates phrase modeling through bigram (n-gram with n=2) probabilities. A bigram model uses the probability $P(T_i \mid T_{i-1})$ the probability of a word given the immediately preceding word and can distinguish, for example, between documents about "stock exchange" in a financial sense versus "exchange of live stock" in an agricultural sense.

For stemming and synonym handling, the framework introduces a statistical translation model with probabilities $P(S_i \mid T_i)$, where $S_i$denotes the user's vocabulary and $T_i$denotes the document vocabulary. This allows morphological variants, synonyms, and even cross-language query terms to be incorporated without altering the underlying formal model. Hiemstra and de Jong (2001) note that "it can be proven

that on-line morphological generation produces the exact same results as the traditional use of a stemmer, if the generated morphological variants are the same words that are conflated by the stemmer."

Check Your Progress – IV

Q1. In the language model for IR, what does the parameter $\lambda_i$represent?

Q2. Fill in the blank: Setting $\lambda_i = 0$for a term is the mathematical equivalent of treating it as a word.

Q3. Who were among the first to propose using statistical language models for information retrieval? (Name at least two.)

Q4. Match the following:

Column A	Column B
Bigram model	Translation probabilities
$\lambda_i = 1$	Phrase modeling
$P(S_i \parallel T_i)$	Mandatory term

11.7 Comparative Analysis of IR Models

The three principal statistical models of IR differ substantially in their formal properties, computational requirements, and practical capabilities. The table below summarizes their key characteristics:

Model Comparison

Feature	Boolean Model	Vector Space Model	Probabilistic/LM Model
Document ranking	No	Yes (continuous)	Yes (probability-based)
Term weighting	No	TF-IDF weights	Mixture parameters
Formal justification of stop words	No	Partial (frequency-based)	Yes ($\lambda_i = 0$)
Relevance feedback	Not native	Yes (Rocchio)	Yes (probability update)
Phrase handling	Via ADJ/NEAR operators	Indirect (term co-occurrence)	Via bigram probabilities
Computational complexity	Low	Moderate	Moderate to High
Output interpretability	Exact match	Cosine/inner product	Probability score

The Boolean model remains in widespread use in commercial database systems due to its efficiency and predictability, but it is inadequate for tasks requiring relevance

ranking. The VSM with TF-IDF weighting represents the standard benchmark and performs robustly across a wide variety of test collections. The probabilistic and language modeling approaches provide a theoretically sound foundation that formally accounts for phenomena such as stop-word behaviour, mandatory terms, and morphological variation that TF-IDF models handle only heuristically.

11.7.1 Precision and Recall

All three models are evaluated using precision and recall, can be defined as follows:

- **Precision** = number of relevant documents retrieved / total number retrieved
- **Recall** = number of relevant documents retrieved / total number relevant in the collection

These measures have an inverse relationship: as more documents are retrieved, recall tends to increase but precision generally decreases. The precision-recall curve is the standard display of retrieval performance, plotting precision against recall at each point where a new relevant document is discovered.

11.8 Practical Implementations and Evaluation

11.8.1 The Inverted Index

All three models rely on the inverted index as their foundational data structure. According to Hull, "the inverted index is a crucial part of almost any retrieval system because it improves search efficiency in a dramatic fashion." Rather than storing a list of terms for each document, the inverted index stores, for each term, a list of all documents (and their positions and frequencies) in which it appears. This allows the system to rapidly identify documents containing any query term and accumulate similarity scores without scanning the full collection.

11.8.2 Latent Semantic Indexing

A significant limitation of VSM is the independence assumption and its inability to handle synonymy and polysemy. Latent Semantic Indexing (LSI) applies the Singular Value Decomposition (SVD) to the term-document matrix, representing both terms and documents in a lower-dimensional subspace of orthogonal indexing variables. According to Hull (n.d.), LSI "is a remarkably successful dimension reduction technique. For search and routing it can actually be used to improve retrieval performance, indicating that there is a lot of noise which can be eliminated from the data." Performance improves steadily with the number of LSI factors included, reaching a peak before declining back to the baseline when all factors are included.

11.8.3 Evaluation Benchmarks

The Text Retrieval Conference (TREC) has been central to the evaluation of IR systems since the early 1990s. It provides standardized test collections, queries, and relevance judgements against which different retrieval models can be objectively compared. Kowalski and Maybury (2002) note that TREC results provide the primary empirical basis for comparing the effectiveness of algorithmic variations in practice. Systems such as SMART, Okapi, and INQUERY have been benchmarked extensively through TREC, and the results inform the continuing refinement of both VSM and probabilistic approaches.

11.9LET US SUM UP

This unit has introduced the three foundational statistical models of information retrieval: the Boolean model, the Vector Space Model, and the probabilistic model (including the language modeling approach). The Boolean model, despite its efficiency, lacks document ranking and term weighting and cannot integrate formal linguistic knowledge. The VSM overcomes these limitations through TF-IDF term weighting and ranked retrieval, supporting relevance feedback via the Rocchio method, though it assumes term independence. The statistical language modeling approach provides a unified theoretical framework in which stop words, mandatory terms, phrase modeling, stemming, and cross-language retrieval are all accommodated as special cases of a single probabilistic model. Pre-processing techniques like tokenization, stop-word removal, stemming, and phrase extraction, prepare text for all three models and their effectiveness is determined by evaluation measures of precision and recall across standardized test collections.

✓ 11.10 ANSWER TO CHECK YOUR PROGRESS

Check Your Progress-I

Ans to Q1. Stop-word removal eliminates semantically weak, high-frequency words that would otherwise match many documents without contributing to relevance discrimination, thereby reducing index size and improving retrieval precision.

Ans to Q2.

Column A	Column B
Porter Algorithm	Morphological normalization
Tokenization	Conversion of text to word sequences
Phrase extraction	Multi-word processing token

Ans to Q3. 30 to 50 percent.

Ans to Q4. False. Stemming "tends to help as many queries as it hurts" and may conflate words with different meanings.

Check Your Progress-II

Ans to Q1. The AND operator corresponds to set intersection.

Ans to Q2. NEAR operator.

Ans to Q3. Any two from: no ranking of documents; no concept of term importance; query is highly sensitive to term choice; no automatic relevance feedback; uncontrolled output size.

Ans to Q4.

Column A	Column B
AND	Set intersection
OR	Set union
NOT	Set complement

Check Your Progress-III

Ans to Q1. IDF stands for Inverse Document Frequency. It measures how rare a term is across the document collection — terms appearing in fewer documents receive a higher IDF weight.

Ans to Q2. Vectors.

Ans to Q3. True.

Ans to Q4. (b) Rocchio Method.

Check Your Progress-IV

Ans to Q1. The parameter $\lambda_i$represents the probability that the i-th query term is "important" that is, drawn specifically from the vocabulary of the relevant document rather than from general language.

Ans to Q2. Stop word.

Ans to Q3. Hiemstra (1998), Miller et al. (1999), and Ponte and Croft (1998).

Ans to Q4.

Column A	Column B
Bigram model	Phrase modeling
$\lambda_i = 1$	Mandatory term
$P(S_i \parallel T_i)$	Translation probabilities

11.11 MODEL QUESTIONS

Short Questions

1. Define information retrieval and distinguish it from a database management system.
2. What are the three standard Boolean operators, and what set operations do they correspond to?
3. Explain the concept of TF-IDF weighting and why it is preferred over raw term frequency.
4. What is relevance feedback, and how does the Rocchio method implement it in the VSM?
5. State the basic language model equation for IR and explain the role of the parameter λ_i.
6. Why does setting $\lambda_i = 0$in the statistical language model formally justify stop-word removal?
7. What is Latent Semantic Indexing, and how does it address the term independence assumption of the VSM?
8. Define precision and recall. Why do they typically have an inverse relationship?
9. Describe the inverted index and explain why it is fundamental to efficient IR.
10. List two advantages of the probabilistic language model over TF-IDF-based ranked retrieval.

Long Answer Questions

1. Compare and contrast the Boolean model and the Vector Space Model of information retrieval. Discuss their respective strengths, limitations, and suitability for different retrieval tasks.
2. Explain how the statistical language modeling approach to information retrieval provides a formal mathematical framework for natural language processing techniques.
3. Discuss the role of term weighting in the Vector Space Model. Describe the components of TF-IDF weightingand explain how these components interact to improve ranked retrieval.

11.12 REFERENCES AND FURTHER READINGS

References

- Hiemstra, D. and de Jong, F. M. G. (2001). "Statistical Language Models and Information Retrieval: Natural Language Processing Really Meets Retrieval." *Glot International*, 5(8), 288–294.

- Hull, D. A. (n.d.). *Information Retrieval Using Statistical Classification*. PhD thesis, Stanford University.
- Kowalski, G. J. and Maybury, M. T. (2002). *Information Storage and Retrieval Systems: Theory and Implementation*, 2nd edition. Kluwer Academic Publishers, Dordrecht.Information-Storage-And-Retrieval-Systems-Theory-And-Impl-2e_Kowalski-GJ-2002.pdf
- Ponte, J. M. and Croft, W. B. (1998). "A language modeling approach to information retrieval." *Proceedings of the 21st ACM Conference on Research and Development in Information Retrieval (SIGIR'98)*, pp. 275–281.
- Salton, G. and Buckley, C. (1988). "Term-weighting approaches in automatic text retrieval." *Information Processing and Management*, 24(5), 513–523.
- Shannon, C. E. (1948). "A mathematical theory of communication." *Bell System Technical Journal*, 27, 379–423, 623–656.
- Hiemstra, D. (2000). "A probabilistic justification for using tf.idf term weighting in information retrieval." *International Journal on Digital Libraries*, 3(2), 131–139.
- Baeza-Yates, R. A. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison-Wesley, London.
- Manning, C. and Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*. MIT Press, Boston.
- Van Rijsbergen, C. J. (1979). *Information Retrieval*, 2nd edition. Butterworths, London.

Further Readings

- Manning, C. D., Raghavan, P. and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.
- Robertson, S. E. and Sparck Jones, K. (1976). "Relevance Weighting of Search Terms." *Journal of the American Society for Information Science*, 27(3), 129–146.
- Croft, W. B., Metzler, D. and Strohman, T. (2010). *Search Engines: Information Retrieval in Practice*. Pearson Education.
- Deerwester, S., Dumais, S., Furnas, G., Landauer, T. and Harshman, R. (1990). "Indexing by Latent Semantic Analysis." *Journal of the American Society for Information Science*, 41(6), 391–407.

*** ***** ***

UNIT 12: NATURAL LANGUAGE PROCESSING IN INFORMATION RETRIEVAL

UNIT STRUCTURE

12.1 Learning Objectives

12.2 Introduction

12.3 Natural Language Processing in Information Retrieval

12.4 NLP Techniques and Tasks

- 12.4.1 Tokenization
- 12.4.2 Stemming and Stemming Algorithms
- 12.4.3 Lemmatization
- 12.4.4 Comparison of Stemming and Lemmatization
- 12.4.5 NLP Dictionaries

12.5 Semantic Analysis

- 12.5.1 Knowledge Base
- 12.5.2 Knowledge Representation
  - 12.5.2.1 Semantic Networks
  - 12.5.2.2 Frames

12.6 Let Us Sum Up

12.7 Answer to Check Your Progress

12.8 Model Questions

12.9 References and Further Readings

12.1 Learning Objectives

After going through this unit, you will be able to:

- Provide a thorough understanding of Natural Language Processing (NLP) and its role in information retrieval systems.
- Familiarise learners with the linguistic phenomena that affect retrieval performance.
- Examine key NLP tasks such as tokenization, stemming, and lemmatization.
- Introduce the concepts of semantic analysis, knowledge bases, and knowledge representation through semantic networks and frames.

12.2 Introduction

Information retrieval systems are designed to respond to user queries by locating and delivering documents whose content matches the user's information need. In practice, users express their information needs using natural language, either as a statement or as part of a dialogue with the system. However, natural language is inherently ambiguous, and this ambiguity frequently causes retrieved documents to be irrelevant to the actual need. The two most significant characteristics of natural language that influence retrieval outcomes are **linguistic variation** (where different words are used to express the same meaning, such as "car," "automobile," and "motorcar") and **linguistic ambiguity** (where the same word carries more than one meaning, such as "crane," which may refer to a bird or a lifting device).

These characteristics affect retrieval at three levels: the syntactic level (the arrangement of words in sentences), the semantic level (the meaning of individual words and combinations of words), and the pragmatic level (the contextual interpretation of language). Natural Language Processing (NLP) offers a principled means of addressing these challenges by analysing and transforming text into forms that retrieval systems can process more accurately and meaningfully.

12.2 Natural Language Processing in Information Retrieval

"According to Vallez and Pedraza-Jimenez (2007) in *Natural Language Processing in Textual Information Retrieval and Related Topics*, NLP is an area of research and application that studies how natural language text entered into a computer system can be manipulated and transformed into a form more suitable for further processing." NLP emerged in 1960 as a sub-field of Artificial Intelligence and Linguistics, with the aim of studying problems related to the automatic generation and understanding of natural language.

The primary function of an NLP-based IR system involves four interconnected tasks. First, during **indexing**, NLP techniques generate document descriptions by identifying terms that best represent the document's content. Second, during **query representation**, the system analyses the user's query and attempts to represent it in a manner consistent with how document content is indexed. Third, during **query processing**, the system matches document descriptions against query representations to retrieve documents with the closest match. Fourth, in the **display of results**, retrieved documents are ranked in order of their relevance to the query.

There are two fundamental approaches to NLP in IR. The **statistical approach** treats all words in a document as index terms and assigns weights based on term frequency, an approach also called the "bag of words" model. While computationally simple, this model fails to account for word ordering, structure, or meaning. The **linguistic approach**, by contrast, encodes explicit linguistic knowledge and analyses documents at the syntactic, semantic, and pragmatic levels, enabling more accurate

interpretation of content. Most modern NLP systems employ a combination of both approaches.

12.3 NLP Techniques and Tasks

Syntactic analysis focuses on the grammatical arrangement of words in sentences. It involves recognising valid sentences and determining their underlying structures through methods such as context-free grammar, transformational grammar, and parsing. From this syntactic foundation, several key NLP tasks emerge that are widely applied in IR systems.

12.3.1 Tokenization

"According to Chowdhury (1999) in *Introduction to Modern Information Retrieval*, tokenization (also known as word segmentation) is the process of breaking up text into words, phrases, symbols, or other meaningful elements, each of which is called a token." The list of tokens produced becomes the input for subsequent processes such as parsing, indexing, and text mining.

Tokenization occurs at the word level and is performed by locating word boundaries, typically identified by whitespace characters (such as spaces or line breaks) or punctuation. In languages that use inter-word spacing, such as those using the Latin alphabet, tokenization is relatively straightforward. However, several edge cases present challenges:

- Hyphenated expressions such as "New York-based" or "State-of-the-art" require careful boundary detection.
- Ancient Greek texts, often written without spaces between words (*scriptio continua*), present significant tokenization difficulties.
- Languages such as Chinese have no word boundaries, requiring specialised segmentation algorithms.

Two widely used tokenization tools are **U-Tokenizer**, an HTTP-based API that segments Chinese and Japanese sentences at word boundaries and also supports English, and **RoboVerdict**, which tokenizes English product reviews to identify similarities.

12.3.2 Stemming and Stemming Algorithms

Stemming is the process of reducing inflected or derivationally related word forms to a common base form, known as the stem. The underlying premise is that the stem carries the core meaning of a concept, while affixes (prefixes and suffixes) serve either syntactic functions or introduce minor modifications to the concept's meaning. For example, the words "automates," "automatic," and "automation" all reduce to the stem "automat," while "fishing," "fished," and "fisher" all reduce to "fish."

"According to Kowalski (2002) in *Information Storage and Retrieval Systems: Theory and Implementation*, stemming is used to improve the efficiency of the information system and to improve recall; conflation is the term frequently used to refer to mapping multiple morphological variants to a single representation or stem." The principal trade-off is that while stemming improves recall by grouping related terms together, it can reduce precision by conflating words with distinct meanings. For instance, the Porter stemmer reduces "operate," "operating," "operation," and "operational" to the single stem "oper," which may cause a query for "operating system" to match a document discussing "operative procedures."

Several stemming algorithms have been developed, the most important of which are described below:

Porter Stemming Algorithm: Developed by Martin F. Porter (1980), this is the most widely used stemming algorithm for English. It operates on a set of rules based on the length of the stem and the nature of the suffix, applying them iteratively across multiple steps to remove common English suffixes. The algorithm tends toward under stemming (leaving some related words with different stems) in order to minimise overstemming errors that reduce precision.

Lovins Stemmer: Developed by Julie Beth Lovins (1968), this was one of the earliest stemming algorithms. It removes the longest possible suffix from a word in a single pass, using a large list of suffix patterns.

Paice-Husk Stemmer: Developed by Chris Paice (1990), this is a more aggressive algorithm than the Porter stemmer, applying iterative rules until no further reductions are possible. It tends toward overstemming, thereby achieving higher recall at the cost of precision. Paice (1994) developed the **Error Rate Relative to Truncation (ERRT)** measure to compare stemming algorithms by quantifying understemming and overstemming through an Understemming Index (UI) and an Overstemming Index (OI).

Kstem (Dictionary Look-Up Stemmer): Used in the INQUERY system, Kstem is a morphological analyser that combines simple stemming rules with dictionary lookup. It attempts to avoid collapsing words with different meanings into the same root; for example, "memorial" and "memorize" are not reduced to a common stem despite morphological similarity.

Successor Stemmers: These algorithms, based on work by Hafer and Weiss, determine word boundaries by measuring the number of distinct letters that follow a given prefix across the vocabulary (known as the "successor variety"). The stem is identified at points where the successor variety is lowest.

An important practical consideration is that certain word categories, such as proper nouns and acronyms, should not be stemmed because their morphological forms are not related to a common conceptual core.

Check Your Progress 1

Q1. What is stemming?

Q2. Name the most commonly used stemming algorithm for English.

Q3. Stemming improves but may reduce in retrieval.

Q4. Match the following:

Stemmer	**Developer**
A. Porter Stemmer	1. Julie Beth Lovins
B. Lovins Stemmer	2. Chris Paice
C. Paice-Husk Stemmer	3. Martin F. Porter

Q5. What does ERRT stand for?

12.3.3 Lemmatization

Lemmatization is the process of grouping together the different inflected forms of a word so that they can be analysed and retrieved as a single item. "According to Wikipedia (cited in Dutta, 2014 in 'Advances Course in Information Storage and Retrieval'), in computational linguistics, lemmatization is the algorithmic process of determining the lemma for a given word." A **lemma** is the canonical, dictionary form of a word. For example, the verb "to walk" may appear as "walk," "walked," "walks," and "walking"; the lemma in each case is "walk." The combination of the base form with its part of speech is often called the **lexeme** of the word.

Lemmatization involves two essential tasks: understanding the **context** in which a word appears, and determining the **part of speech** of the word within that context. These requirements make lemmatization a complex process. For instance, the word "meeting" may function either as a noun (as in "our last meeting") or as a verb (as in "We are meeting tomorrow"), and only contextual analysis can determine the correct lemma. Lemmatizers require a complete vocabulary (lexicon) and a morphological analysis component, and are therefore considerably more resource-intensive than stemmers.

12.3.4 Comparison of Stemming and Lemmatization

While both stemming and lemmatization aim to normalise word forms, they differ substantially in their methodology, accuracy, and practical application. The table below summarises the key distinctions:

Dimension	**Stemming**	**Lemmatization**
Method	Applies rule-based suffix/prefix removal without contextual analysis	Uses vocabulary and morphological analysis with contextual consideration
Context awareness	No context considered	Context and part of speech are considered
Output	Stem (may not be a valid word)	Lemma (always a valid dictionary word)
Example	"better" may remain "better" (no	"better" correctly maps to "good"

	mapping)	
Speed	Faster, simpler to implement	Slower, requires more computational resources
Knowledge required	Language-specific rules	Complete vocabulary and morphological analysis
Accuracy	Lower; errors of overstemming and understemming	Higher; more linguistically precise
Application	Suitable for large-scale retrieval where high precision is not critical	Preferred in systems requiring accurate linguistic analysis

A fundamental limitation of stemmers is that they operate on a single word in isolation and cannot distinguish between words that have different meanings depending on their part of speech. Lemmatizers, in principle, select the appropriate lemma based on context, though this advantage comes at the cost of implementation complexity.

12.3.5 NLP Dictionaries

NLP dictionaries are structured lexical resources that support both stemming and lemmatization processes. They store morphological variants of words and allow systems to look up the appropriate root or base form. Systems such as Retrieval Ware and INQUERY employ dictionary-based components as part of their stemming infrastructure: the INQUERY system's Kstem algorithm uses six major data files including a lexicon, a supplemental word list, an exceptions list, a direct conflation table, a country-nationality mapping file, and a proper nouns list to control the stemming process.

Beyond stemming support, NLP dictionaries serve as the backbone for semantic disambiguation. Resources such as **WordNet**, a large lexical database of English developed at Princeton University, group words into sets of cognitive synonyms called **synsets**, each expressing a distinct concept. Synsets are interlinked by conceptual-semantic and lexical relations, making WordNet a useful tool for both computational linguistics and information retrieval. Unlike a conventional thesaurus, WordNet labels the semantic relations among words (such as synonymy, hypernymy, and hyponymy) rather than merely grouping words by similarity of meaning.

Check Your Progress 2

Q1. What is a lemma?

Q2. Which is faster to implement: a stemmer or a lemmatizer?

Q3. The word "walked" has the lemma.

Q4. True or False: Lemmatization considers the part of speech of a word before determining its base form.

Q5. What is WordNet?

12.4 Semantic Analysis

Semantic analysis constitutes the most complex phase of language processing. In this phase, semantic information is added to the structural representations produced during syntactic analysis. While syntactic analysis determines whether a string of text is a grammatically valid sentence and describes its structural arrangement, semantic analysis draws on knowledge of word meanings, sentence structure, and world knowledge to stipulate the meaning of words, phrases, sentences, and texts, as well as their purpose and contextual implications. The central challenge in semantic analysis is eliminating the ambiguities that purely structural consideration cannot resolve. For example, the sentence "John was reading a book in the bank" is syntactically valid regardless of whether "bank" refers to a financial institution or a riverbank; only semantic knowledge about the context can disambiguate the intended meaning.
Two fundamental concepts underpin semantic analysis in NLP-based IR systems: the **knowledge base** and **knowledge representation**.

12.4.1 Knowledge Base

"According to Chowdhury (1999) in *Introduction to Modern Information Retrieval*, a knowledge base consists of various kinds of knowledge, for instance, it can be a general knowledge about the world, or knowledge about a specific domain." Knowledge bases are useful to NLP systems for processing and interpreting the meanings of natural language statements and for eliminating semantic ambiguities. However, most knowledge bases cover only specific domains and are created by relatively small groups of knowledge engineers; their creation and maintenance are cost-intensive, particularly as domains evolve.

Several widely used knowledge bases merit attention:

- **OpenCyc** is described as the world's largest general knowledge base and commonsense reasoning engine. The version released in June 2012 (OpenCyc 4.0) contained 239,000 concepts and 2,093,000 facts, licensed under the Apache License.
- **WordNet** is a lexical database of English that groups nouns, verbs, adjectives, and adverbs into synsets interlinked by conceptual-semantic and lexical relations. It is freely available for download and is widely used in NLP and computational linguistics.
- **Freebase** is a large collaborative knowledge base of structured metadata harvested from many sources, including community contributions. It was developed by the American software company Metaweb and has been publicly available since March 2007.

- **DBpedia** is a community effort to extract structured information from Wikipedia and make it available on the Web. The English DBpedia knowledge base describes approximately 4.0 million entities classified in a consistent ontology.

12.4.2 Knowledge Representation

"According to Chowdhury (1999) in *Introduction to Modern Information Retrieval*, knowledge representation (also known as semantic representation) is an area that deals with representing knowledge about the world in general or domain-specific knowledge in a form that a computer system can utilise to solve complex tasks, such as having a dialogue in a natural language." Knowledge representation involves creating an internal representation from natural language statements; this representation is not limited to the language of the input text and can be used for further processing in tasks such as query matching, database creation, and text processing.

"According to Kemp (1988) in *Computer-Based Knowledge Retrieval*, the two major purposes of knowledge representation are: (i) to help people understand the system they are working with, and (ii) to enable the system to process the representation." The greatest challenge in this area lies not in gathering knowledge, but in representing and structuring it appropriately, searching it efficiently, and using it to infer further knowledge. Examples of knowledge representation formalisms include semantic networks, frames, conceptual dependency, case grammar, and ontologies. The following sections discuss semantic networks and frames.

12.4.2.1 Semantic Networks

"According to Sowa (1987) in 'Semantic Networks', a semantic network is a network that represents semantic relations between language units, consisting of a directed or undirected graph with vertices (nodes) and edges (arcs)." Both nodes and arcs carry labels. Typically, nodes represent language units such as words, phrases, word senses, sentences, or documents, while edges represent relationships between them in terms of co-occurrence, collection membership, syntactic dependency, or lexical similarity. The networks are described as "semantic" because they encode the meaning of text through explicit relationships between language units.

Semantic networks are proposed as knowledge representations because they enable the storage of language units alongside their interrelations, supporting a variety of inference and reasoning processes. They are particularly useful for forming a conceptual schema of a domain. Two fundamental properties characterise semantic networks:

1. **Inheritance hierarchy:** This refers to the taxonomy of concepts denoted by common nouns. Hierarchies are called inheritance hierarchies because lower-level nodes inherit properties from nodes above them in the hierarchy. For

example, in a network containing the nodes "Penguin is-a Bird" and "Bird is-a Vertebrate," it can be inferred that a Penguin is a Vertebrate.

2. **Intersecting search:** This refers to a method of finding a connection between two concepts by searching through the network for a path linking them.

A simple semantic network might contain nodes such as "Penguin," "Bird," and "Vertebrate," connected by arcs labelled "is-a," along with additional arcs such as "Bird has-part Wings." From the set of facts embedded in the network, new facts can be inferred: from "Penguin is-a Bird" and "Bird is-a Vertebrate," the network allows the inference that "Penguin is-a Vertebrate."

12.4.2.2 Frames

The frame is one of the most widely used schemes for knowledge representation in NLP systems. "According to Chowdhury (1999) in *Introduction to Modern Information Retrieval*, the frame concept was first proposed by Marvin Minsky in his article 'A Framework for Representing Knowledge' (1974) as a basis for understanding visual perception, natural language communication, and other complex behaviours."

A frame is an artificial intelligence data structure used to divide knowledge into substructures by representing stereotyped situations. Frames provide a structure within which new data are interpreted in terms of concepts acquired through previous experiences, with the organisation of knowledge facilitating expectation-driven reasoning by directing the system to look for elements expected in a given context.

The central representational mechanism within a frame is the **slot**. Each piece of information about a particular frame is held in a slot, and each slot may hold a specific value. A frame contains information about how to use the frame, what to expect next, what other frames are likely to apply in a given context, and how to respond when expectations are not met.

For example, a "Student" frame might contain slots for Name, Sex, Age, Date of Birth, Home Address, Course Name, and Skill. A specific instance of this frame might have the values: Name = "John," Sex = "Male," Date of Birth = "05.12.1993," Course Name = "MSLIS," Skill = "Computer."

Frames may also be linked such that the value of one slot points towards a new frame. For instance, the value "MSLIS" in the "Course Name" slot of the Student frame might point to a separate "Courses" frame containing detailed information about that programme. This linking mechanism allows complex, hierarchically organised knowledge structures to be built up from individual frames. It is also significant to note that because frames are structurally based, it is possible to generate a semantic network given a set of frames, even though frames lack explicit arcs between nodes.

Check Your Progress 3

Q1. Who proposed the concept of frames in knowledge representation?

Q2. What are the two fundamental properties of semantic networks?

Q3. In a frame, each individual piece of information is held in a .

Q4. Match the following knowledge bases with their descriptions:

Knowledge Base	Description
A. WordNet	1. Crowd-sourced structured data extracted from Wikipedia
B. DBpedia	2. Lexical database grouping English words into synsets
C. OpenCyc	3. General knowledge base with 239,000 concepts (2012)

Q5. True or False: A semantic network is a linear data structure.

12.5 LET US SUM UP

Natural language is the most practical and natural means through which users interact with information retrieval systems, yet its inherent linguistic variation and ambiguity represent the most significant barriers to effective retrieval. NLP addresses these barriers by applying a range of computational techniques at the syntactic, semantic, and pragmatic levels.

Among the key syntactic-level tasks, tokenization breaks text into meaningful units for further processing; stemming normalises morphological variants to a common stem, improving recall at the possible cost of precision; and lemmatization provides a linguistically precise alternative to stemming by returning valid dictionary forms with the aid of contextual and part-of-speech analysis. NLP dictionaries, including lexical resources such as WordNet, support both normalisation tasks and semantic disambiguation.

At the semantic level, knowledge bases such as OpenCyc, WordNet, Freebase, and DBpedia provide the factual and conceptual grounding that NLP systems require to interpret natural language statements. Knowledge representation formalisms, particularly semantic networks and frames, provide structured means of encoding this knowledge in a form that computer systems can process, reason over, and apply to the task of retrieving information accurately and meaningfully in response to user queries.

ANSWER TO CHECK YOUR PROGRESS

Check Your Progress 1

Ans to Q1. Stemming is the process of reducing inflected or derivationally related word forms to a common base or root form, called a stem, so that related words map to the same representation.

Ans to Q2. The Porter Algorithm, developed by Martin F. Porter in 1980.

Ans to Q3. Recall; precision.

Ans to Q4. A-3, B-1, C-2.

Ans to Q5. Error Rate Relative to Truncation, a measure developed by Paice to compare stemming algorithm performance.

Check Your Progress 2

Ans to Q1. A lemma is the canonical, base, or dictionary form of a word; for example, "walk" is the lemma of "walked," "walks," and "walking."

Ans to Q2. A stemmer is faster and simpler to implement than a lemmatizer.

Ans to Q3. "walk."

Ans to Q4. True. Lemmatization requires understanding the part of speech and context of a word.

Ans to Q5. WordNet is a large lexical database of English developed at Princeton University that groups words into synsets and interlinks them through semantic and lexical relations.

Check Your Progress 3

Ans to Q1. Marvin Minsky, in his 1974 article "A Framework for Representing Knowledge."

Ans to Q2. Inheritance hierarchy and intersecting search.

Ans to Q3. Slot.

Ans to Q4. A-2, B-1, C-3.

Ans to Q5. False. A semantic network is a graph structure consisting of nodes and arcs, not a linear structure.

12.6 MODEL QUESTIONS

Short Answer Questions

1. Define Natural Language Processing (NLP).
2. What is tokenization in the context of NLP?
3. State any two differences between stemming and lemmatization.
4. What is a synset in WordNet?
5. Define a knowledge base.
6. What is meant by "inheritance hierarchy" in a semantic network?
7. Name the two fundamental techniques of NLP applied in IR.
8. What is meant by "linguistic variation" in natural language?
9. What is a slot in the context of frames?
10. Who developed the Porter Stemming Algorithm and in which year?

Long Answer Questions

1. Discuss the role of Natural Language Processing in Information Retrieval. How do NLP-based IR systems handle the tasks of indexing, query representation, and result display?
2. Explain the Porter Stemming Algorithm with its structure and rules. What are its advantages and limitations compared to the Lovins and Paice-Husk stemmers?
3. Compare and contrast stemming and lemmatization in detail. Under what circumstances would a system designer prefer one over the other?
4. What is a knowledge base? Discuss the major knowledge bases used in NLP systems, including WordNet, OpenCyc, Freebase, and DBpedia, with their key characteristics.
5. Explain semantic networks as a form of knowledge representation. How do inheritance hierarchy and intersecting search facilitate reasoning and inference in NLP systems?
6. Describe frames as a knowledge representation formalism. Illustrate with an example how frames are structured and how linked frames can represent complex knowledge domains.
7. Discuss the two fundamental approaches to NLP (statistical and linguistic) and evaluate their respective strengths and weaknesses in the context of information retrieval.

12.7 REFERENCES AND FURTHER READINGS

References

- Chowdhury, G. G. *Introduction to Modern Information Retrieval*. London: Library Association Publishing, 1999.
- Dutta, B. "Natural Language Processing." In *Advances Course in Information Storage and Retrieval*, Module 11. DRTC, Indian Statistical Institute, Bangalore.
- Kowalski, G. J. *Information Storage and Retrieval Systems: Theory and Implementation*. 2nd ed. New York: Kluwer Academic Publishers, 2002.
- Manning, C. D., Raghavan, P., and Schutze, H. *Introduction to Information Retrieval*. Cambridge: Cambridge University Press, 2008.
- Vallez, M. and Pedraza-Jimenez, R. "Natural Language Processing in Textual Information Retrieval and Related Topics." *Hipertext.net*, no. 5, 2007.
- Porter, M. F. "An Algorithm for Suffix Stripping." *Program*, 14(3), pp. 130-137, 1980.
- Lovins, J. B. "Development of a Stemming Algorithm." *Translation and Computational Linguistics*, 11(1), pp. 22-31, 1968.
- Paice, C. D. "Another Stemmer." *SIGIR Forum*, 24(3), pp. 56-61, 1990.

- Minsky, M. "A Framework for Representing Knowledge." MIT-AI Laboratory Memo 306, 1974.
- Sowa, J. F. "Semantic Networks." In *Encyclopedia of Artificial Intelligence*, edited by S. Shapiro. New York: Wiley, 1987. Available at: http://www.jfsowa.com/pubs/semnet.htm
- Kemp, D. *Computer-Based Knowledge Retrieval*. London: Aslib, 1988.
- Salton, G. *Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer*. Reading, MA: Addison-Wesley, 1989.
- Allan, J. "NLP for IR: Natural Language Processing for Information Retrieval." *NAACL-ANLP Language Technology Joint Conference*, Washington, USA, 2000.
- Liddy, E. D. "Natural Language Processing for Information Retrieval and Knowledge Discovery." Available at: https://www.ideals.illinois.edu/bitstream/handle/2142/26000/LiddyNatural.pdf

Further Readings

- Goker, A. and Davies, J. (eds.) *Information Retrieval: Searching in the 21st Century*. Chichester: Wiley, 2009.
- Garbham, A. *Artificial Intelligence: An Introduction*. London: Routledge and Kegan Paul, 1988.
- Miller, G. A. "WordNet: A Lexical Database for English." *Communications of the ACM*, 38(11), pp. 39-41, 1995.
- Antoniou, G. and Harmelen, F. van. *A Semantic Web Primer*. London: MIT Press, 2004.
- Wilks, Y. and Brewster, C. "Natural Language Processing as a Foundation of the Semantic Web." *Foundations and Trends in Web Science*, 1(3-4), pp. 199-327, 2009.
- Hafer, M. A. and Weiss, S. F. "Word Segmentation by Letter Successor Varieties." *Information Storage and Retrieval*, 10(11-12), pp. 371-385, 1974.

*** ***** ***

UNIT 13: WEB-BASED INFORMATION RETRIEVAL

UNIT STRUCTURE

13.1 Learning Objectives

13.2 Introduction

13.3 World Wide Web

13.4 Users' Interaction and Search Behaviour on the Web

13.5 Search Engines

13.6 Web Directories

13.7 Semantic Web and Its Role in Information Retrieval

13.8 Let Us Sum Up

13.9 Answer to Check Your Progress

13.10 Model Questions

13.11 References and Further Readings

13.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- Describe the structure of the World Wide Web and its document model.
- Explain the naming conventions used to identify web documents.
- Analyse user interaction patterns and search behaviour on the web.
- Distinguish between types of search engines and their respective features.
- Explain the purpose and functioning of web directories.
- Understand the concept of the Semantic Web and its role in improving IR.

13.2 INTRODUCTION

The emergence of the World Wide Web (WWW) has fundamentally transformed the landscape of information retrieval (IR). Unlike classical IR systems, which operated within controlled and relatively homogeneous environments, web-based IR operates across a distributed, dynamic, and heterogeneous information space. We will learn the structural foundations of the Web, the document model and naming conventions that make retrieval possible, the patterns of user interaction and search behaviour, the architecture and typology of search engines, the role of web directories, and the contribution of the Semantic Web to modern IR practice. Together, these topics illuminate both the theoretical underpinnings and the practical challenges of retrieving information in the web environment.

13.3 WORLD WIDE WEB

13.3.1 Origin and Architecture of the WWW

The World Wide Web is a vast, globally distributed system founded on client-server architecture. "According to Madalli in *Web Based Information Retrieval* (Module 14), 'The World Wide Web (WWW) is a huge system based on client-server architecture with millions of distributed servers worldwide. Here, every server maintains a collection of documents and each document is stored as a file although documents can also be generated on request.'" The Web originated as a project at CERN, the European Particle Physics Laboratory in Geneva, designed to provide shared document access to a geographically dispersed research community. Its simplicity of design, based on the concept of linking documents from different locations without requiring centralised control, proved to be instrumental in its exponential growth.

A browser serves as the client application through which users interact with web servers. The browser is responsible for properly rendering and displaying documents, as well as accepting user inputs, typically in the form of link selections, and subsequently fetching and displaying the requested documents. This client-server model provides the foundational interaction pattern for all web-based information retrieval activity.

13.3.2 The Document Model

In the web environment, all information is represented in the form of documents. These documents vary significantly in complexity, ranging from simple ASCII text files to collections of executable scripts that run automatically when downloaded into a browser. A document may also contain references to other documents in the form of hyperlinks, enabling non-linear navigation across the information space.

Hyper Text Mark-up Language (HTML) is the basic building block of web documents. "According to Madalli in *Web Based Information Retrieval*, HTML is 'a mark-up language that provides keywords to provide structure into various sections.'" HTML enables structuring of content through elements such as headings, tables, lists, forms, and embedded multimedia. In addition to structural elements, HTML provides rendering instructions to the browser, governing how content is displayed to the user. More advanced web documents may also integrate scripting languages, such as JavaScript, to enable dynamic content generation.

The significance of the document model in IR lies in how web documents are indexed, represented, and retrieved. Unlike structured database records, web documents are semi-structured or entirely unstructured. "According to Kowalski and Maybury in *Information Storage and Retrieval Systems: Theory and Implementation* (2nd ed., 2002), items in an Information Retrieval System can be 'composed of text including numeric and date data, images, audio, video and other multi-media objects.'"

This diversity of document types presents IR systems with significant challenges in representation and retrieval.

13.3.3 Naming: URIs, URLs, and URNs

The WWW employs a standardised naming scheme to identify and locate documents. "According to Berners-Lee et al. (1998), as cited in Madalli's*Web Based Information Retrieval*, the Web uses 'Uniform Resource Identifiers or simply URIs' as its naming scheme." The URI encompasses two related concepts: the Uniform Resource Locator (URL) and the Uniform Resource Name (URN).

A URL identifies a document by specifying its location, including the protocol, host server, and file path. A URN, by contrast, provides a persistent, location-independent reference to a document, functioning as a true identifier rather than a location pointer. This distinction is significant from an IR perspective. "According to Kowalski and Maybury (2002), 'from a library and large information system perspective it is far more efficient to refer to items by name rather than location.'" Location-based identifiers such as URLs become unreliable over time as documents are moved or servers are decommissioned, whereas URNs are designed to maintain persistence beyond the organisational life of the creating entity.

Check Your Progress 1

Q1. What does WWW stand for?

Q2. HTML stands for _____.

Q3. Match the following:

Column A	Column B
(a) URL	(i) Location-independent identifier
(b) URN	(ii) Location-based identifier
(c) URI	(iii) Naming scheme that includes URL and URN

Q4. Where did the World Wide Web originate as a project?

13.4 USERS' INTERACTION AND SEARCH BEHAVIOUR ON THE WEB

13.4.1 Nature of Web Users

The user population in web-based IR differs markedly from that of classical IR systems. In traditional library-based or database IR environments, the user group was relatively homogeneous and consisted primarily of academics, researchers, subject experts, and librarians who were familiar with controlled vocabulary searches and

formal query construction. The web, however, has democratised information access, drawing in a vastly heterogeneous user population that includes novices across a wide spectrum of subject backgrounds and varying levels of computer and web literacy.

This shift in user profile has had direct implications for IR system design. "According to Madalli in *Web Based Information Retrieval*, 'the user interaction with the different interface of search engines largely depends on the skills of users to frame the search query in such a manner so as to bridge the gap between the way a programmer has indexed a particular content under what keyword and the actual requirement of the user.'" This gap between the user's expressed query and the system's indexing vocabulary is a central challenge in web-based IR.

13.4.2 Web Search Behaviour Patterns

Searching for information on the web can be a difficult and time-consuming experience owing to the sheer volume and heterogeneity of available content. Web content differs from traditional library resources in that it is heterogeneous, networked, and available in multiple media types, including text, hypertext, images, audio, and video. Information creation on the web is dynamic and operates beyond physical and institutional boundaries.

In a traditional library setting, a reference librarian acts as an intermediary between the user and the information resources, analysing the user's query and directing retrieval. On the web, users must themselves perform this intermediary function, while search engines return all documents matching the query keywords without the benefit of a trained professional's interpretation. The user must subsequently filter and evaluate the returned documents to identify relevant materials, a task that demands a degree of information literacy often absent in novice users.

Several studies have explored web search behaviour. "According to Peiling Wang et al. (1998) in 'An exploratory study of user searching of the World Wide Web: a holistic approach,' presented at the 61st Annual Meeting of the American Society for Information Science, user search behaviour on the web involves complex cognitive and contextual factors beyond simple keyword matching." "According to Mansourian (2004) in 'Similarities and differences between Web search procedure and searching in the pre-web information retrieval systems,' published in *Webology*, the transition from pre-web to web-based searching introduces notable differences in user strategies, query formulation, and relevance judgment."

13.4.3 Classic IR versus Web-Based IR

The distinction between classical IR and web-based IR (WBIR) is important to understanding how retrieval strategies have evolved. In classical IR systems, digital contents from online and offline databases and Online Public Access Catalogues (OPACs) were stored in a structured manner, making search and retrieval more predictable. With the emergence of the web, digital information flooded the information

landscape in an unstructured, uneven, and heterogeneous manner. Web-based IR systems evolved to address these challenges by maintaining regularly updated indexes of web documents and providing user-friendly interfaces designed with the novice user in mind.

Check Your Progress 2

Q1. In a traditional library environment, who acts as a bridge between the user and information?

Q2. OPAC stands for.

Q3. Which of the following best describes web content?

(a) Homogeneous and structured

(b) Heterogeneous and networked

(c) Static and limited

(d) Print-only and controlled

Q4. Name one key challenge in web-based IR related to the user profile.

13.5 SEARCH ENGINES

13.5.1 Definition and Function

Search engines are computer programs that search for keywords entered by users and return a list of documents in which those keywords are found. More specifically, they are services designed to search contents on the web. Their functionality extends beyond simple keyword matching, and their technical architecture differs considerably across types. "According to Croft, Metzler, and Strohman (2010) in *Search Engines: Information Retrieval in Practice*, search engines represent the primary practical realisation of IR principles in the web environment."

"According to Kowalski and Maybury (2002), 'the objective of the search capability is to allow for a mapping between a user's specified need and the items in the information database that will answer that need.'" This mapping process is mediated by various algorithms and data structures that form the core of search engine architecture.

13.5.2 Types of Search Engines

Search engines can be categorised into four main types based on their indexing methodology and retrieval mechanism:

- **Crawler-based search engines** use automated programs called crawlers or spiders to systematically browse and index web pages by following hyperlinks

from one page to another. They are useful when the user has a specific keyword in mind. However, for general search topics they may return numerous irrelevant results. A notable example is AltaVista.

- **Human-powered directories** rely on human editors who review and classify websites into hierarchical categories. They are particularly effective for general topic searches, as the editorial classification helps converge the search and deliver refined responses. DMOZ (the Open Directory Project) is a prominent example.
- **Hybrid search engines** combine crawler-based indexing with human-edited directory results to provide both breadth and precision. Google is the most widely used hybrid search engine.
- **Meta-search engines** do not maintain their own index but instead aggregate results from multiple independent search engines at a single interface. They are efficient for determining the availability of information on a topic across different sources. Dogpile is a well-known example.

13.5.3 Features of Search Engines

Modern search engines provide a range of standard features to facilitate effective retrieval. The principal features include:

- **Boolean search**, which allows the combination of keywords using logical operators AND, OR, and NOT, enabling users to define precise retrieval conditions. This can be implemented through operator syntax or through menu-driven options such as "all of the words" as in Google.
- **Proximity search**, which retrieves documents in which two or more specified words occur within a defined number of words of each other. Google supports proximity search through the AROUND operator.
- **Field search**, which restricts the search to a specific part of the document, such as the title, using operators like "intitle:" in Google.
- **Phrase search**, which retrieves documents containing an exact sequence of words, entered in double quotation marks.
- **Limiting search**, which allows users to filter results by date or file type.

"According to Kowalski and Maybury (2002), 'proximity is used to increase the precision of a search. If the terms COMPUTER and DESIGN are found within a few words of each other then the item is more likely to be discussing the design of computers than if the words are paragraphs apart.'"

13.5.4 Advanced Search Engines and Applications

Advanced search engines go beyond text-based retrieval to address specific and specialised information needs. The following represent significant categories of advanced search applications:

- **General search with advanced options:** Google's advanced search interface allows users to perform highly specific queries by combining field restrictions, date limits, language filters, and file type specifications.
- **Reverse image search:** Engines such as TinEye analyse image content directly, making images searchable by content rather than file name or user-defined tags alone.
- **Similar image search:** Engines such as GazoPa identify visually similar images based on features such as texture, colour, and structure.
- **Invisible web search:** The Complete Planet search engine is designed to retrieve content from databases that are not accessible to general crawlers. This type of content, often referred to as the Deep Web, is invisible to standard search engines because it is not linked through publicly accessible hyperlinks.
- **Semantic search:** Engines using semantic search algorithms, such as Yummly, retrieve results based on the meaning and context of search terms rather than their literal occurrence.

Check Your Progress 3

Q1. Which type of search engine uses automated programs to browse and index web pages?

Q2. Meta-search engines maintain their own index. (True/False)

Q3. Match the following:

Column A	Column B
(a) TinEye	(i) Meta-search engine
(b) Dogpile	(ii) Hybrid search engine
(c) Google	(iii) Reverse image search engine

Q4. What does "Deep Web" refer to in the context of search engines?

13.6 WEB DIRECTORIES

A web directory, also known as a link directory, is a type of directory available on the WWW whose purpose is to provide categorised links to other websites. Unlike search engines, web directories do not display results based on user-entered keywords. Instead, they maintain a classified list of websites organised according to subject categories and subcategories. "According to Boswell, as cited in Madalli's *Web Based Information Retrieval*, a web directory classifies websites 'according to class and subclass'" rather than indexing individual pages.

An important distinguishing feature of web directories is that their classification is based on the complete website rather than on individual pages within it. Websites may be included in a directory through two mechanisms: submission by the site owner

followed by an editorial review and approval process, or direct addition by the directory's editorial staff.

Web directories are particularly useful when a user is exploring a broad subject domain and seeks guidance through an organised classification structure. In this sense, they function similarly to the subject classification schemes used in physical libraries. Among the most notable web directories are Yahoo! Directories, one of the oldest and most extensive directories on the Web, and the Open Directory Project (DMOZ), which is a human-edited directory maintained voluntarily by a global community of editors.

The principal difference between a web directory and a search engine may be summarised as follows:

Feature	**Web Directory**	**Search Engine**
Indexing method	Human-edited, subject classification	Automated crawling of pages
Retrieval basis	Browse by category	Keyword-based query
Scope	Entire websites	Individual pages
Currency	Periodic manual updates	Continuous automated updates
User requirement	Exploratory browsing	Specific information need

Despite their organisational value, web directories have faced significant decline with the rise of powerful hybrid search engines. Maintaining an editorially curated directory at the scale of the modern web presents a formidable challenge, which partly explains the discontinuation of large directories such as DMOZ in 2017.

13.7 SEMANTIC WEB AND ITS ROLE IN INFORMATION RETRIEVAL

13.7.1 Ontology: The Conceptual Foundation

The Semantic Web builds upon the conceptual framework of ontology. The concept of ontology originated more than two thousand years ago in philosophy, specifically from Aristotle's theory of categories, whose original purpose was to provide a comprehensive categorisation of all existing things in the world. In more recent decades, ontologies have been adopted in fields such as Library and Information Science (LIS), Artificial Intelligence (AI), and Computer Science (CS) as the primary means for describing how classes of objects are correlated and for categorising document resources.

"According to Gruber (1993) in 'A translation approach to portable ontology specifications' in *Knowledge Acquisition*, ontology is defined as 'an explicit specification of a conceptualization.'" This definition was subsequently extended by Studer, Benjamins, and Fensel (1998) in *Knowledge Engineering: Principles and Methods*, who defined ontology as "a formal, explicit specification of a shared conceptualisation," incorporating the notion of shared understanding and formal relations among concepts.

The key features of ontologies relevant to IR include: providing a shared understanding of subject domains; facilitating the sharing of domain knowledge between human users and automated agents; supporting the organisation and navigation of websites; and improving the accuracy of web searches by enabling generalisation and specialisation of information.

13.7.2 The Semantic Web

The Semantic Web represents an extension of the existing web in which information is given well-defined meaning, enabling computers and people to work in closer cooperation. Rather than treating web content as a collection of text strings to be matched statistically, the Semantic Web aims to represent the meaning of that content in a machine-readable form. "According to W3C's *Semantic Web Made Easy*, the Semantic Web is an extension of the current web in which information is given well-defined meaning."

From a technical standpoint, Semantic Web data is addressed by URLs and retrieved in the form of Semantic Web Documents (SWD). "According to Madalli in *Web Based Information Retrieval*, 'the Semantic Web is an extensive collection of static or dynamic semantic web documents (SWD) distributed over the Web-space.'" Meta-information attached to each ontology supports searching and matching operations, and enables the retrieval of ontologies stored in an ontology registry, providing a compact representation for efficient search and reuse.

13.7.3 Semantic Search in IR

Semantic search is a combined effort of Semantic Web technologies and search engine technologies, designed to resolve complex queries, support automatic clustering, and manage large volumes of web documents. In contrast to conventional keyword-based search, semantic search retrieves results based on the meaning, context, and definition of terms. This approach addresses the fundamental limitation of statistical retrieval methods, which evaluate relevance based on term frequency and co-occurrence rather than conceptual meaning.

"According to Dini (2004) in 'NLP technologies and the semantic web: risks, opportunities and challenges,' published in *Intelligenza Artificiale*, natural language processing technologies play a central role in bridging the gap between syntactic string matching and semantically informed retrieval." Semantic search engines such as Swoogle and Watson represent early implementations of this paradigm, employing ontological reasoning and structured knowledge representations to enhance retrieval accuracy.

The broader contribution of the Semantic Web to IR may be understood at three levels. At the document level, semantic annotation enriches web documents with machine-readable metadata that conveys meaning beyond the literal text. At the query level, ontologies enable query expansion through generalisation and specialisation,

allowing a search for "automobile" to encompass related terms such as "car" or "vehicle" without explicit user instruction. At the system level, shared ontologies facilitate interoperability between disparate IR systems, enabling more coherent cross-domain and cross-language retrieval.

Check Your Progress 4

Q1. Who defined ontology as "an explicit specification of a conceptualization"?

Q2. Semantic search retrieves results based on rather than literal keyword occurrence.

Q3. Name two semantic search engines mentioned in the literature.

Q4. The Semantic Web was proposed as an extension of the existing web. (True/False)

13.8 LET US SUM UP

This unit has examined the key dimensions of web-based information retrieval. The World Wide Web, founded on client-server architecture, uses HTML documents and a URI-based naming system comprising URLs and URNs to enable global document identification and access. User interaction on the web is characterised by a heterogeneous population whose search behaviours differ substantially from those in classical IR environments, placing greater demands on individual information literacy. Search engines, which can be classified as crawler-based, human-powered, hybrid, or meta-search engines, provide the primary interface between users and the vast web information space, and incorporate features such as Boolean, proximity, phrase, and field search. Web directories offer an editorially organised alternative to keyword-based retrieval, particularly for exploratory searches. Finally, the Semantic Web and its foundation in ontological knowledge representation offer a principled approach to overcoming the limitations of statistical IR by enabling meaning-based retrieval through semantic search technologies.

13.9 ANSWER TO CHECK YOUR PROGRESS

Check Your Progress 1

Ans to Q1. WWW stands for World Wide Web.

Ans to Q2. HTML stands for Hyper Text Mark-up Language.

Ans to Q3. (a) - (ii); (b) - (i); (c) - (iii).

Ans to Q4. The World Wide Web originated as a project at CERN, the European Particle Physics Laboratory in Geneva.

Check Your Progress 2

Ans to Q1. The reference librarian acts as a bridge between the user and information in a traditional library environment.

Ans to Q2. OPAC stands for Online Public Access Catalogue.

Ans to Q3. (b) Heterogeneous and networked.

Ans to Q4. A key challenge is the diversity of the user population, as many web users are novices with limited skills for framing effective search queries.

Check Your Progress 3

Ans to Q1. Crawler-based search engines use automated programs (crawlers or spiders) to browse and index web pages.

Ans to Q2. False. Meta-search engines do not maintain their own index; they aggregate results from multiple other search engines.

Ans to Q3. (a) - (iii); (b) - (i); (c) - (ii).

Ans to Q4. The Deep Web refers to web content stored in databases that is not accessible or indexed by standard search engines because it is not linked through publicly accessible hyperlinks.

Check Your Progress 4

Ans to Q1. Gruber (1993) defined ontology as "an explicit specification of a conceptualization."

Ans to Q2. Semantic search retrieves results based on meaning, context, and definition.

Ans to Q3. Swoogle and Watson are two semantic search engines mentioned in the literature.

Ans to Q4. True.

13.10MODEL QUESTIONS

Short Answer Questions

1. Define the World Wide Web and describe its basic architecture.
2. What is a URL? How does it differ from a URN?
3. Distinguish between a crawler-based search engine and a human-powered directory.
4. What are the main features common to most modern search engines?

5. Define a web directory and explain how websites are included in it.
6. What is ontology? Name the scholar who first formally defined it in the context of knowledge representation.
7. What is semantic search, and how does it differ from keyword-based search?
8. What is the Deep Web? Which type of search engine is designed to access it?

Long Answer Questions

1. Explain the document model of the World Wide Web in detail. Discuss the role of HTML in structuring web documents and the significance of URIs in web-based information retrieval.
2. Discuss user interaction and search behaviour on the web. How does the typical web user's approach to search differ from that of a user of classical IR systems? What challenges arise from this difference for IR system design?
3. Describe the four types of search engines. For each type, explain its mechanism of operation, identify its strengths and limitations, and provide an example.
4. Write a detailed note on the Semantic Web. Explain the concept of ontology, the role of semantic search, and the contribution of Semantic Web technologies to improving information retrieval on the web.
5. Compare and contrast web directories and search engines as tools for web-based information retrieval. Under what circumstances would each be more appropriate for an information seeker?

13.11 REFERENCES AND FURTHER READINGS

References

- Antoniou, G. and Harmelen, F. van. (2004). *A Semantic Web Primer*. London: MIT Press.
- Basu, A. and Paul, M. (2015). A Case Study on Semantic Web Search Engines. In *Proceedings of Libraries in Next Era (LiNE)*. India. ISBN: 978-81-930849-0-8.
- Berners-Lee, T. et al. (1998). Uniform Resource Identifiers (URI): Generic Syntax. IETF RFC 2396.
- Croft, W. B., Metzler, D. and Strohman, T. (2010). *Search Engines: Information Retrieval in Practice* (p. 88). Reading: Addison-Wesley.
- Dini, L. (2004). NLP technologies and the semantic web: risks, opportunities and challenges. *Intelligenza Artificiale*, 1(1), pp. 67-71.
- Dutta, B., Chatterjee, U. and Madalli, D. P. (2013). From Application Ontology to Core Ontology. In *Proceedings of International Conference on Knowledge Modelling and Knowledge Management (ICKM 2013)*. Bangalore, India. ISBN: 978-93-5137-765-8.

- Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2), pp. 199-220.
- Gulli, A. and Signorini, A. (2005). The indexable web is more than 11.5 billion pages. In *Proceedings of the 14th International Conference on World Wide Web* (pp. 902-903). New York: ACM.
- Kowalski, G. J. and Maybury, M. T. (2002). *Information Storage and Retrieval Systems: Theory and Implementation* (2nd ed.). New York: Kluwer Academic Publishers.
- Madalli, D. P. (n.d.). Web Based Information Retrieval. Module 14. New Delhi: CEC/UGC.

Further Readings

- Baeza-Yates, R. and Ribeiro-Neto, B. (2011). *Modern Information Retrieval: The Concepts and Technology behind Search* (2nd ed.). Harlow: Addison-Wesley.
- Berners-Lee, T., Hendler, J. and Lassila, O. (2001). The Semantic Web. *Scientific American*, 284(5), pp. 34-43.
- Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1-7), pp. 107-117.
- Jacso, P. (2004). Testing the calculation of a realistic h-index in Google Scholar, Scopus, and Web of Science for F. W. Lancaster. *Library Trends*, 53(4), pp. 557-578.
- Manning, C. D., Raghavan, P. and Schutze, H. (2008). *Introduction to Information Retrieval*. Cambridge: Cambridge University Press.

*** ***** ***

UNIT 14 THE SEMANTIC WEB

UNIT STRUCTURE

14.1 Learning Objectives
14.2 Introduction
14.2.1 The Semantic Web Vision
14.3 XML: Extensible Markup Language
14.3.1 Salient Features of XML
14.3.2 Issues and Limitations of XML
14.4 RDF: Resource Description Framework
14.4.1 The RDF Triplet
14.4.2 Salient Features of RDF
14.5 RDF Schema (RDFS)
14.5.1 Key RDFS Constructs
14.5.2 Differences Between RDF and RDFS
14.5.3 Limitations of RDFS
14.6 Ontology, Logic, and Description Logics
14.6.1 What is an Ontology?
14.6.2 Logic and Ontology Languages
14.6.3 Components of Description Logics
14.6.4 Standard DL Reasoning Tasks
14.7 Web Ontology Language (OWL)
14.7.1 OWL Full
14.7.2 OWL DL
14.7.3 OWL Lite
14.8 P`roof Layer and Trust Layer
14.8.1 The Proof Layer
14.8.2 The Trust Layer
14.8.3 The Complete Semantic Web Stack: A Synthesis
14.9 Let us sum up
14.10 Answer to Check your progress
14.11 Model Questions
14.12 Further Reading and References

14.1 Learning Objectives

After studying this chapter, students will be able to:

1. Explain the concept and vision of the Semantic Web.
2. Define RDF and construct RDF triplets.

3. Distinguish between RDF and RDF Schema (RDFS).
4. Explain ontologies and their relevance to knowledge organisation

14.2 INTRODUCTION

The World Wide Web, since its inception by Tim Berners-Lee in 1989, has primarily served as a vast repository of human-readable documents. Browsers render HTML pages for human view, and search engines index text for human queries. However, the web's fundamental limitation has always been that computers can find data but cannot understand it. A machine can locate a word in a document but cannot determine the context of it.

In 2001, Berners-Lee, James Hendler, and Ora Lassila introduced the vision of the Semantic Web in a landmark article in Scientific American. They described it as an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation. This vision marked a paradigm shift: from a web of documents to a web of data.

For professionals in Library and Information Science (LIS), the Semantic Web is not merely a technical curiosity. It aligns profoundly with the discipline's foundational goals: organising knowledge, enabling precise retrieval, and providing structured access to information. Cataloguing, classification, controlled vocabularies, and subject headings are all precursors to the semantic technologies discussed in this chapter. The Semantic Web represents a digital evolution of the principles that have guided librarianship for over a century.

14.2.1 The Semantic Web Vision

The Semantic Web envisions a web where each piece of data is associated with a formal definition, enabling software agents to automatically navigate resources, combine information from diverse sources, and draw logical inferences. Berners-Lee described this architecture as a layered stack, often called the Semantic Web Layer Cake, where each layer depends on the standards and technologies beneath it.
The principal layers of this stack, from the foundation upwards, are:

- URI / IRI — Universal identifiers for every resource
- XML — Structured syntax for data interchange
- RDF and RDFS — Graph-based data model and schema vocabulary
- OWL — Rich ontology language for complex knowledge representation
- Logic / Rules — Inference engines and rule languages
- Proof — Transparent reasoning chains
- Trust — Authentication and credibility of sources

14.3 XML: eXtensible Markup Language

XML, or eXtensible Markup Language, serves as the syntactic foundation of the Semantic Web. Developed by the World Wide Web Consortium (W3C) and standardised in 1998, XML was designed to store and transport structured data in a format that is both human-readable and machine-parsable. Unlike HTML, which uses a fixed set of tags for presentation, XML allows users to define their own tags to describe the nature of the data contained within.

In the Semantic Web architecture, XML functions as a common interchange format. It provides a consistent syntax upon which higher-level semantic languages such as RDF and OWL are built. Although XML itself carries no inherent meaning it is purely syntactic it offers a disciplined and extensible structure for encoding information.

14.3.1 Salient Features of XML

XML possesses several properties that made it attractive as the syntactic layer of the Semantic Web:

- Self-Describing: XML tags describe the nature of their content. A tag such as <Author> communicates meaning to a human reader, unlike binary formats.
- Extensible: Users define their own vocabulary of tags, making XML applicable to any domain, from library catalogues to medical records.
- Platform-Independent: XML is plain text and can be processed on any operating system or programming language without proprietary software.
- Hierarchical Structure: XML uses a tree-based document model with a single root element, enabling clear parent-child relationships between data elements.
- Namespace Support: Namespaces allow elements from different vocabularies to coexist in a single document without naming conflicts — critical when integrating data from multiple sources.
- Schema Validation: Document Type Definitions (DTDs) and XML Schema Definitions (XSD) allow documents to be validated against a formal structure.
- Unicode Support: XML supports the full Unicode character set, enabling true internationalisation of data.
- Separation of Data and Presentation: When combined with XSLT (Extensible Stylesheet Language Transformations), XML cleanly separates data content from its display format.

14.3.2 Issues and Limitations of XML

Despite its utility, XML has significant limitations that prevent it from fulfilling the full vision of the Semantic Web. These limitations necessitated the development of higher-level languages such as RDF.

- **Lack of Shared Meaning:** XML syntax defines structure but not semantics. The tag <Bank> in one document might mean a financial institution, in another a riverbank. Without an agreed-upon vocabulary, machines cannot interpret the intended meaning of elements.
- **Verbosity:** XML syntax is highly repetitive and verbose, resulting in large file sizes that can be inefficient for large-scale data exchange.
- **No Reasoning Capability:** XML provides no mechanism for inferring new knowledge from existing data. It is a passive data container, not a knowledge representation language.
- **No Relationship Semantics:** While XML can represent hierarchical relationships through nesting, it cannot express arbitrary relationships between entities, such as "authorOf" or "isPartOf", in a machine-interpretable way.
- **Interoperability Challenges:** The same information can be represented in multiple valid XML structures, making it difficult to integrate data from different sources without prior agreement on the schema.
- **No Linked Data Support:** XML documents are self-contained. There is no native mechanism for linking data across documents or across the web.

14.4 RDF: Resource Description Framework

The Resource Description Framework (RDF) is a W3C standard for representing information about resources in the World Wide Web. Introduced in 1999 and revised in 2004 and 2014, RDF moves beyond XML's purely syntactic capabilities to provide a machine-interpretable data model. RDF introduces the concept of semantics to web data by providing a standard way to describe the properties and relationships of resources using a graph-based model.

In RDF, all knowledge is expressed as a collection of simple statements, each consisting of three components: a subject, a predicate, and an object. This tripartite structure, known as an RDF triplet or triple, forms the atomic unit of all semantic description in the Semantic Web.

14.4.1 The RDF Triplet

An RDF triplet expresses a single fact or relationship about a resource. Each component has a specific role:

Component	Description	Represented As
Subject	The resource being described	URI or Blank Node
Predicate	The property or relationship being asserted	URI (from a defined vocabulary)

Component	Description	Represented As
Object	The value of the property or the related resource	URI, Blank Node, or Literal

In RDF, subjects and predicates are expressed as URIs (Uniform Resource Identifiers), which ensures that every entity and relationship is globally unique and unambiguous. Objects can be URIs (pointing to another resource) or literals (plain values such as strings, dates, or numbers). This use of URIs is what makes RDF suitable for linked data — different data sources can refer to the same entity using the same URI.

Multiple RDF triplets together form a directed graph, where nodes represent subjects and objects, and directed edges (arrows) represent predicates. This graph model enables complex networks of relationships to be represented, traversed, and reasoned over.

14.4.2 Salient Features of RDF

- URI-based IDs: Every resource in RDF is identified by a URI, providing a globally unique, persistent identifier. This is the foundation of linked data.
- Graph Data Model: RDF data forms a labelled, directed graph. This is more flexible than the hierarchical model of XML, allowing many-to-many relationships.
- Open-World Assumption: Unlike relational databases, RDF assumes that the absence of a statement does not imply it is false. This is crucial for incomplete or distributed knowledge bases.
- Multiple Serialisations: RDF can be serialised in multiple formats: RDF/XML, Turtle, N-Triples, JSON-LD. This flexibility aids interoperability.
- Mergeability: RDF graphs from completely different sources can be merged into a single graph without conflicts, provided URIs are used consistently.

14.5 RDF Schema (RDFS)

RDF Schema, commonly abbreviated as RDFS, is a semantic extension of RDF that provides a vocabulary for describing classes and properties of RDF resources. While RDF provides the mechanism to make statements about resources, RDFS provides the vocabulary for describing the types of resources and the relationships between them. In essence, RDFS acts as a meta-vocabulary or basic type system for RDF.

Published as a W3C Recommendation in 2004, RDFS introduces the notion of classes (groupings of similar resources) and class hierarchies (subclass relationships).

It also allows properties to be constrained in terms of the types of resources they can apply to (domain) and the types of values they can take (range).

14.5.1 Key RDFS Constructs

The following table summarises the principal vocabulary terms defined in RDFS:

RDFS Term	Purpose and Example
rdfs:Class	Defines a class (a type of resource). Example: :Personrdf:typerdfs:Class
rdfs:subClassOf	States that one class is a subtype of another, supporting inheritance. Example: :Librarianrdfs:subClassOf :Professional
rdfs:subPropertyOf	States that one property is more specific than another. Example: :hasMobilePhonerdfs:subPropertyOf :hasContactInfo
rdfs:domain	Specifies the class of resources that a property can be applied to. Example: :authorOfrdfs:domain :Person
rdfs:range	Specifies the class of values a property can take. Example: :authorOfrdfs:range :Document
rdfs:label	Provides a human-readable name for a resource
rdfs:comment	Provides a human-readable description of a resource
rdfs:seeAlso	Indicates a resource that provides further information about the subject

14.5.2 Differences Between RDF and RDFS

The distinction between RDF and RDFS is a common point of confusion for students. The following comparison clarifies their respective roles:

Dimension	RDF	RDFS
Primary Purpose	Describe individual resources (instances)	Describe classes and properties (schema)
Level of Abstraction	Instance level — specific facts about things	Schema/Class level — general types and relationships
Type of Statement	This resource has this property with this value	This property applies to resources of this class

Dimension	RDF	RDFS
Hierarchy Support	No native hierarchy	Supports class and property hierarchies (subClassOf)
Reasoning Support	Very limited	Basic inference through subclass/property chains
Analogy in LIS	Individual catalogue records	The cataloguing schema or metadata standard itself

14.5.3 Limitations of RDFS

Although RDFS represents a significant improvement over plain RDF, it remains insufficient for many knowledge representation tasks. Its principal limitations are:

- No cardinality constraints: RDFS cannot express that a person must have exactly one date of birth, or that a book may have up to five keywords.
- No disjointness: RDFS cannot formally state that two classes are mutually exclusive (e.g., a resource cannot simultaneously be a Periodical and a Monograph).
- No enumerated classes: A class cannot be defined as the closed set of specific instances.
- No property characteristics: There is no way to declare a property as transitive (e.g., isAncestorOf), symmetric (e.g., isMarriedTo), or functional (e.g., hasBirthDate).
- No equivalence: RDFS cannot state that two different URIs refer to the same concept.

These limitations motivated the development of a richer language. The Web Ontology Language (OWL), discussed in Section 4.6, was designed specifically to address them.

14.6 ONTOLOGY, LOGIC, AND DESCRIPTION LOGICS

14.6.1 What is an Ontology?

The term ontology originates in philosophy, where it refers to the branch of metaphysics concerned with the nature and categories of being. In computer science and information science, the term has been adapted to mean a formal, explicit specification of a shared conceptualisation. This widely cited definition, offered by Thomas Gruber in 1993, encapsulates four key properties:

- Formal: The ontology is expressed in a language with a well-defined syntax and formal semantics, making it machine interpretable.

- Explicit Concepts, their properties, and their relationships are explicitly stated, leaving no ambiguity.
- Shared: The ontology represents a consensus view of a domain, agreed upon by a community, not an individual perspective.
- Conceptualisation: An abstract model of the world, describing what kinds of things exist and how they relate to one another.

An ontology typically consists of the following components: classes (or concepts), properties (or roles), instances (or individuals), axioms (logical statements that must hold), and constraints. Together, these elements constitute a formal model of knowledge in a domain.

14.6.2 Logic and Ontology Languages

For an ontology to support automated reasoning, it must be expressed in a formal logic. The Semantic Web uses Description Logics (DL) as the logical underpinning of its ontology languages. DL is a family of formal knowledge representation languages designed to balance expressiveness with computational tractability that is, the ability to perform reasoning within a reasonable amount of time.

Predicate logic offers maximum expressiveness but is undecidable (reasoning may never terminate). Propositional logic is decidable but too weak to represent rich knowledge. Description Logics occupy a carefully chosen middle ground, offering sufficient expressive power for real-world knowledge bases while guaranteeing that standard reasoning tasks can be completed in finite time.

14.6.3 Components of Description Logics

A Description Logic knowledge base is divided into two distinct components:

Component	Description	Example
TBox (Terminological Box)	Contains schema-level knowledge: class definitions, property declarations, and axioms describing general relationships between classes.	"Every Medical Doctor is a Professional" a general rule about all members of a class
ABox (Assertional Box)	Contains instance-level knowledge: specific facts about individual entities in the domain.	"Dr. Priya Sharma is a Medical Doctor" — a fact about a specific individual

In addition to the TBox/ABox distinction, Description Logics work with three types of objects:

- Concepts (Classes): Sets of individuals sharing certain properties. Example: the concept 'RareManuscript' in a library ontology.
- Roles (Properties): Binary relationships between individuals. Example: 'isStoredIn' relates a manuscript to a storage location.
- Individuals: Named instances belonging to concepts. Example: 'Manuscript_Ashburnham_1' is an individual of class 'RareManuscript'.

14.6.4 Standard DL Reasoning Tasks

A DL reasoning engine can perform several standard inference tasks over a knowledge base:

1. Satisfiability: Determines whether a concept is logically consistent — i.e., whether it is possible for any individual to belong to it without contradiction.
2. Subsumption: Determines whether one class is necessarily a subclass of another — whether all members of class A must also be members of class B.
3. Classification: Automatically places each concept in its correct position in the class hierarchy based on its definition.
4. Instance Checking: Determines whether a specific individual must belong to a specific class, given the axioms in the knowledge base.
5. Realisation: Finds the most specific class or classes to which a given individual belongs.

14.7 Web Ontology Language (OWL)

The Web Ontology Language (OWL) is a formal ontology language developed by the W3C and standardised as a recommendation in 2004 (OWL 1) and subsequently revised in 2009 (OWL 2). OWL is built upon the RDF/RDFS framework and extends it with substantially richer vocabulary for expressing complex knowledge, including class equivalences, disjointness, cardinality restrictions, and property characteristics.

OWL is grounded in Description Logics, which provides it with a formal, well-understood semantics and guarantees decidability (in its restricted sublanguages). OWL is considered the most expressive standardised ontology language on the Semantic Web and forms the basis of numerous applied ontologies in biomedicine, library science, geospatial data, and beyond.

The W3C defined three sublanguages of OWL 1, each offering a different balance between expressive power, computational complexity, and ease of use. These are OWL Full, OWL DL, and OWL Lite.

14.7.1 OWL Full

OWL Full is the most expressive of the three sublanguages. It encompasses the full syntax and semantics of RDF, allowing any OWL vocabulary to be applied to any RDF construct. In OWL Full, a class can simultaneously be treated as an instance (an individual), meaning the distinction between classes and individuals is completely dissolved.

Key characteristics of OWL Full:

- Maximally expressive — all OWL and RDF constructs can be freely combined.
- No separation of classes and individuals — a class can itself be an instance of another class.
- Undecidable — no general-purpose reasoning algorithm can guarantee termination for all OWL Full ontologies.
- No complete reasoners exist — tools cannot guarantee to find all valid conclusions.
- Appropriate for scenarios where maximum flexibility is required and automated reasoning is not essential.

14.7.2 OWL DL

OWL DL (Description Logic) is the sublanguage of OWL that corresponds most directly to its Description Logic foundations. It restricts OWL Full by enforcing a strict separation between classes, properties, and individuals — each URI may play only one of these roles. This restriction, while reducing flexibility compared to OWL Full, restores computational decidability and completeness.

Key characteristics of OWL DL:

- High expressivity — supports all OWL constructs within the constraints of Description Logic.
- Strict separation of classes, properties, and individuals.
- Decidable — reasoning is guaranteed to terminate.
- Complete reasoners available — tools such as Pellet, HermiT, and FaCT++ can find all valid inferences.
- Corresponds to the SHOIN(D) Description Logic, one of the most expressive decidable DLs.
- Recommended for most practical ontology engineering tasks requiring automated reasoning.

14.7.3 OWL Lite

OWL Lite is the most restricted sublanguage of OWL. It was designed to provide a simpler, more easily implementable language for users who primarily need a classification hierarchy and simple constraint checking. OWL Lite supports only a subset of the constructs available in OWL DL, most notably restricting cardinality to only 0 or 1.

Key characteristics of OWL Lite:

- Lowest expressive power of the three sublanguages.
- Decidable and computationally simpler than OWL DL.
- Cardinality restrictions limited to values of 0 or 1 (not arbitrary numbers).
- Easier to migrate from existing thesauri or taxonomies, such as library classification schemes.
- Suitable for straightforward hierarchical knowledge organisation tasks.

14.8 Proof Layer and Trust Layer

The uppermost layers of the Semantic Web architecture are the Proof Layer and the Trust Layer. These layers move beyond knowledge representation and reasoning to address the fundamental questions of credibility and verification: How do we know that a particular conclusion is logically valid? How do we know whether we can trust the source of information we are using?

14.8.1 The Proof Layer

The Proof Layer sits immediately above the Logic/Rules layer in the Semantic Web stack. Its purpose is to make the reasoning process transparent and verifiable. When an automated agent draws a conclusion for example, determining that a particular journal article belongs to a specific subject category the Proof Layer requires that agent to produce an explicit trace of the reasoning steps that led to that conclusion.

The Proof Layer serves several important functions:

- Transparency of Reasoning: The explicit proof trace shows every inference step, making automated deductions understandable and auditable.
- Verification: A separate agent or system can independently verify the proof by checking each step against the axioms and rules that were applied.
- Trust Foundation: Proofs are necessary for agents to trust each other's conclusions in automated multi-agent workflows.
- Accountability: In legal, medical, or archival contexts, proof traces provide accountability for automated decisions.

14.8.2 The Trust Layer

The Trust Layer is the topmost layer of the Semantic Web stack and addresses the social dimension of knowledge on the web: not merely whether a conclusion follows logically, but whether the source of the knowledge is credible and authoritative. This is an especially critical concern in an open, decentralised web where anyone can publish data using Semantic Web standards.

The Trust Layer relies primarily on digital signatures — cryptographic mechanisms that allow a data publisher to attach a verifiable signature to their data, confirming its authenticity and integrity. A consuming agent can verify the signature using the publisher's public key, confirming that:

- The data was published by the claimed source (authentication).
- The data has not been altered since it was signed (integrity).

The Trust Layer also encompasses broader concepts such as:

- Reputation systems and trust policies that allow agents to assign different levels of credibility to different sources.
- Provenance tracking, which records the origin and history of data.
- Public Key Infrastructure (PKI), which provides the cryptographic framework for digital signatures.

14.8.3 The Complete Semantic Web Stack: A Synthesis

Layer	Technology / Standard	Purpose
Trust	Digital Signatures, PKI	Verify authenticity and credibility of data sources
Proof	Proof languages, reasoning traces	Explicit, verifiable reasoning chains
Logic / Rules	SPARQL, SWRL, RIF	Inference and rule-based reasoning over RDF graphs
Ontology	OWL (Full, DL, Lite)	Rich knowledge representation with formal semantics
Schema	RDFS	Class hierarchies and property definitions
Data Model	RDF + Triplets	Graph-based model for facts and relationships
Syntax	XML + Namespaces	Structured, machine-parsable data exchange format

Layer	Technology / Standard	Purpose
Identifiers	URI / IRI	Globally unique identifiers for every resource

14.9 LET US SUM UP

This chapter has presented a comprehensive account of the Semantic Web, from its philosophical vision to its technical architecture. The key concepts covered are as follows:

The Semantic Web Vision: Proposed by Berners-Lee in 2001, the Semantic Web extends the existing web by giving data well-defined meaning, enabling machine-to-machine communication and automated reasoning. It is built as a layered stack of standards.

XML: Provides the syntactic foundation. While XML is self-describing, extensible, and platform-independent, it lacks semantic capabilities. The same data can be modelled in multiple incompatible ways, and XML cannot support reasoning.

RDF: Introduces a graph-based data model built on triplets (Subject–Predicate–Object). All resources are identified by URIs, enabling globally unambiguous identification and linked data. RDF supports mergeability and the open-world assumption.

RDFS: Adds a schema layer to RDF, introducing classes, subclass hierarchies, and domain/range constraints. RDFS enables basic inheritance-based inference but lacks cardinality, disjointness, and rich property characteristics.

Ontologies: Formal, explicit specifications of shared conceptualisations. They extend RDFS with complex axioms, enabling automated reasoning and consistency checking. Ontologies are the formal successors to library classification schemes and thesauri.

Description Logics: The logical foundation of OWL. DL knowledge bases are divided into the TBox (terminological axioms about classes) and the ABox (assertions about individuals). Standard reasoning tasks include satisfiability, subsumption, classification, and instance checking.

OWL: The most expressive W3C ontology standard. OWL Full is maximally expressive but undecidable. OWL DL provides decidable, complete reasoning with high expressivity. OWL Lite is the simplest sublanguage, suitable for classification hierarchies.

Proof and Trust Layers: The topmost layers address the credibility and verification of semantic data. The Proof Layer provides transparent, auditable reasoning chains. The Trust Layer uses digital signatures and PKI to authenticate data sources.

Check your progress

1. What is the Semantic Web?
2. What is the primary role of XML in the Semantic Web?
3. What is RDF?
4. Why is a logic layer needed in the Semantic Web?

14.10 ANSWER TO CHECK YOUR PROGRESS

1. An extension of the current Web in which information is given machine-readable meaning, enabling computers to understand and process data intelligently.
2. XML provides the common syntactic foundation a standard way to structure and exchange data across the Web.
3. RDF is a W3C standard for representing and exchanging information about resources on the Web in a machine-readable format.
4. To enable automated inference allowing machines to derive new facts from existing ones using formal rules and axioms.

14.11 Model Questions

1 Define the Semantic Web. In what ways does it differ from the conventional World Wide Web? Discuss with reference to the limitations of HTML.

2 Explain the structure of an RDF triplet. Construct five RDF triplets relevant to a library catalogue record for a monograph of your choice.

3 Compare and contrast RDF and RDF Schema. What additional capabilities does RDFS provide over plain RDF, and what are its remaining limitations?

4 What is an ontology? Trace the parallels between formal ontologies and traditional Library and Information Science knowledge organisation systems such as classification schemes and thesauri.

5 Explain the TBox and ABox distinction in Description Logics. Provide original examples from a library or information science domain.

6 Critically compare OWL Full, OWL DL, and OWL Lite. Under what circumstances would an ontology engineer choose OWL Lite over OWL DL? Justify your answer.

7 What is the purpose of the Proof Layer in the Semantic Web stack? How does it differ from the Trust Layer? Discuss the significance of both layers for the management of digital library systems.

8 Drawing on the complete Semantic Web Layer Cake, explain how the layers build upon one another to realise Berners-Lee's vision of a machine-understandable web.

14.12 REFERENCESAND FURTHER READING

- Antoniou, G., & van Harmelen, F. (2008). A Semantic Web Primer (2nd ed.). MIT Press.
- Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The Semantic Web. Scientific American, 284(5), 34–43.
- Gruber, T. R. (1993). A translation approach to portable ontology specifications. Knowledge Acquisition, 5(2), 199–220.
- Hitzler, P., Krotzsch, M., & Rudolph, S. (2009). Foundations of Semantic Web Technologies. CRC Press.
- Manola, F., & Miller, E. (2004). RDF Primer. W3C Recommendation. https://www.w3.org/TR/rdf-primer/

*** ***** ***