

**Scheme of Instruction & Examination**  
**B. E. – HONORS –AI&DS**  
**V SEMESTER**

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination		Credits
			L	T	P / D	Contact Hours / week	CIE	SEE	
<b>Theory Courses</b>									
1	H1-CS	Natural Language Processing	3	0	0	3	40	60	3
2	H2-CS	Natural Language Processing Lab	0	0	2	2	40	60	1
<b>Practical / Laboratory Courses</b>									
3	H3-CS	Technical Seminar						60	1
<b>Total Credits</b>						<b>05</b>	<b>80</b>	<b>180</b>	<b>05</b>

**VI SEMESTER**

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination		Credits
			L	T	P / D	Contact Hours / week	CIE	SEE	
<b>Theory Courses</b>									
1	H4-CS	No SQL Databases	3	0	0	3	40	60	3
<b>Practical / Laboratory Courses</b>									
2	H5-CS	Comprehensive Viva-Voce						60	1
3	H6-CS	SWAYAM/GO Lang(Self Learning)	0	0	3	3	40	60	3
<b>Total Credits</b>						<b>06</b>	<b>80</b>	<b>180</b>	<b>07</b>

**VII SEMESTER**

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination		Credits
			L	T	P / D	Contact Hours / week	CIE	SEE	
<b>Theory Courses</b>									
1	H7-CS	Research Methodology	2	0	0	2	40	60	2
<b>Practical / Laboratory Courses</b>									
2	H8-CS	Project Work	0	0	4	4	40	60	4
<b>Total Credits</b>						<b>06</b>	<b>80</b>	<b>120</b>	<b>06</b>

# **V SEMESTER**

# NATURAL LANGUAGE PROCESSING

<b>Semester V</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>Credits</b>
<b>Subject code:H1-CS</b>	<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

<b>Course Objectives:</b>	<b>Course Outcomes:</b>
<ul style="list-style-type: none"><li>➤ To understand to the fundamental concepts and tools of Natural Language Processing (NLP) for text data.</li><li>➤ To equip students with the knowledge and skills required for text and sentence embedding.</li><li>➤ To enable students to classify and cluster text data using a variety of techniques.</li><li>➤ To use of strong foundation in RNNs and their applications in sequence processing.</li><li>➤ To prepare students for real-world NLP applications and challenges.</li></ul>	<ol style="list-style-type: none"><li>1. Understand and implement essential text pre-processing techniques and gain proficiency in N-gram language models</li><li>2. Understand to represent text data using various techniques</li><li>3. Use text classification and clustering methods</li><li>4. Use the concepts of sequence processing with Recurrent Neural Networks (RNNs) and their applications</li><li>5. Use of NLP techniques to various practical applications</li></ol>

## UNIT I

**Introduction to NLP:** Text pre-processing, Regular Expression, tokenization, Stemming, Minimum Edit distance

**N-Gram Language Model:**Intro. N-Gram, N-Gram probability estimation and perplexity, Smoothing technique (Laplace/good Turing/Kneser-Ney/Interpolation)

## UNIT II

**Text Representation:**Bag-of-word:Tf/IDF,Count vector, Vector space Model, Latent semantic Analysis, Word embedding: Word2Vec, Glove, FastText,Sentence embedding Technique: Doc2Vec

## UNIT III

**Text classification and clustering:** The text classification problem, Feature Selection, Naive Bayes text classification, k- nearest neighbors, Support vector Machine, Flat Clustering, K-means algorithm,Hierarchical clustering

**Morphology and part of speech tagging:** Morphology, Part of speech Tagging - Rule-Based Part of Speech Tagging, Markov Models - Hidden Markov Models, Viterbi algorithms, Maximum Entropy Models.

## UNIT IV

**Sequence Processing with Recurrent Networks:** Simple Recurrent Neural Networks, Applications of Recurrent Neural Networks, Deep Networks: Stacked and Bidirectional RNNs, Managing Context in RNNs: LSTMs and GRUs, Words, Subwords and Characters

## UNIT V

**NLP Applications:**Information Extraction, Introduction to Named Entity Recognition and Relation Extraction,Question Answering,Text Summarization

**Dialog System:** Machine Translation

## **TEXTBOOKS**

1. “Speech and Language Processing:  
An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition”, D. Jurafsky and J. Martin III Edition draft
2. “Foundations of Statistical Natural Language Processing”, C. Manning and H. Schutze, MIT Press, 1999

## **REFERENCE BOOKS**

1. Data Analysis from Scratch with Python Step By Step Guide, Peters Morgan, II Edition, AI Sciences
2. Python for Data Analysis: A Step-By-Step Guide to Master the Basics of Data Science and Analysis in Python Using Pandas, NumpyAndIpython, Volume 2, Andrew Park,

## NATURAL LANGUAGE PROCESSING USING PYTHON LAB

<b>Semester V</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>Credits</b>
<b>Subject code: H2-CS</b>	<b>0</b>	<b>0</b>	<b>2</b>	<b>1</b>

<b>Course Objectives:</b>	<b>Course Outcomes:</b>
<ul style="list-style-type: none"><li>➤ Develop the skills to preprocess and clean text data (NLP) tasks.</li><li>➤ Explore feature selection techniques</li><li>➤ Understand the principles behind word embedding</li><li>➤ To implement text processing with LSTM</li><li>➤ To perform NLP application in Sentiment Analysis</li></ul>	<ol style="list-style-type: none"><li>1.Acquire Text Processing Proficiency</li><li>2. Understand N-gram Language Model</li><li>3. Understand Feature Extraction and Representation</li><li>4. Implement Word Embedding Competence</li><li>5. Develop NLP application in Sentiment Analysis</li></ol>

### List of Programs

1. Basic Text Processing operation on text document.
2. Implement N-gram Language model
3. Write a program to extract features from text
4. Implement word embedding using Word2Vec/Glove/fastText
5. Implement LSA and Topic model.
6. Implementation text classification using Naïve Bayes, SVM.
7. Implementation of K-means Clustering algorithm on text.
8. Implement PoS Tagging on text
9. Implement text processing with neural network
10. Implement text processing with LSTM
11. Implement HMM/CRF on sequence tagging task
12. Develop any one NLP application Sentiment Analysis
  - Chatbot
  - Text Summarization Track
  - Machine Translation
  - Question/Answering

## TECHNICAL SEMINAR

Semester V	L	T	P	Credits
Subject code: H3-CS	0	0	0	1

There shall be a seminar presentation in the Honors degree program. For the seminar, the student shall collect the information on a specialized topic and prepare a technical report, showing his understanding over the topic, and submit to the department before presentation. The report and the presentation shall be evaluated by the departmental committee consisting of Head of the department, seminar supervisor and a senior faculty member. The seminar shall be evaluated for 60 marks. A student shall acquire 1 credit only when he/she secures 24 marks on aggregate out of 60 marks allocated. If he/she fails to get 24 marks out of 60, he/she has to re-register for the seminar as and when it is offered.

### Topic Selection:

- Choose a topic that aligns with the field of computer science and engineering.
- Ensure your chosen topic is sufficiently advanced to meet the requirements of the Honors Program.

### 2. Proposal Submission:

- Submit a well-defined proposal outlining your chosen topic, research objectives, methodology, and expected outcomes.
- The proposal should be approved by your advisor or a designated faculty member before proceeding.

### 3. Research and Preparation:

- Conduct extensive research on your chosen topic, using scholarly articles, books, online resources, and academic databases.
- Gather relevant data, conduct experiments and analyze results.
- Consult with your advisor or mentor regularly for guidance and feedback.

### 4. Presentation Format:

- Your presentation should be well-structured, logical, and engaging.
- A typical presentation should last 20-30 minutes, followed by a Q&A session.
- Visual aids such as slides, diagrams, and multimedia elements are encouraged.

### 5. Content and Organization:

- Begin with an introduction that sets the context and outlines the objectives of your seminar.
- Organize your presentation logically, covering the background, related work, your research methodology, findings, and conclusions.
- Highlight the significance and potential applications of your research.

### 8. Evaluation:

- Your seminar will be evaluated based on the depth of research, clarity of presentation, your ability to answer questions, and your overall understanding of the topic.
- The evaluation panel consists of faculty members.

# **VI SEMESTER**

## NoSQL DATABASE

<b>Semester VI</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>Credits</b>
<b>Subject code -H4-CS</b>	<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

Course Objectives:	Course Outcomes:
<ul style="list-style-type: none"> <li>➤ To understand the basic concepts and the applications of database systems. To master the basics of SQL and construct queries using SQL</li> <li>➤ To understand the relational database design principles</li> <li>➤ To become familiar with the basic issues of transaction processing and concurrency control</li> <li>➤ To become familiar with database storage structures and access techniques</li> </ul>	<ol style="list-style-type: none"> <li>1. Explain and compare different types of NoSQL Databases</li> <li>2. Compare and contrast RDBMS with different NoSQL databases.</li> <li>3. Demonstrate the detailed architecture and performance tune of Document-oriented NoSQL databases.</li> <li>4. Explain performance tune of Key-Value Pair NoSQL databases.</li> <li>5. Apply Nosql development tools on different types of NoSQL Databases.</li> </ol>

### UNIT I

Overview and History of NoSQL Databases. Definition of the Four Types of NoSQL Database, The Value of Relational Databases, Getting at Persistent Data, Concurrency, Integration, Impedance Mismatch, Application and Integration Databases, Attack of the Clusters, The Emergence of NoSQL, Key Points.

### UNIT II

Comparison of relational databases to new NoSQL stores, MongoDB, Cassandra, HBASE, Neo4j useand deployment, Application, RDBMS approach, Challenges NoSQL approach, Key-Value and Document Data Models, Column-Family Stores, Aggregate-Oriented Databases. Replication and sharding, MapReduce on databases.

### UNIT III

NoSQL Key/Value databases using MongoDB, Document Databases, Document oriented Database  
Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms.

### UNIT IV

Column-oriented NoSQL databases using Apache HBASE, Column-oriented NoSQL databases using Apache Cassandra, Architecture of HBASE, Column-Family Data Store Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems.



## UNIT V

NoSQL Key/Value databases using Riak, Key-Value Databases, Key-Value Store, Key-Value Store Features, Consistency, Transactions, Query Features, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, User Profiles, Preferences, Shopping Cart Data, Relationships among Data, Multioperation Transactions, Query by Data, Operations by Sets. Graph NoSQL databases using Neo4j.

## TEXT BOOKS

1. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Sadalage, P. & Fowler, Wiley Publications, I Edition, 2019.

## WEB REFERENCES

1. <https://www.ibm.com/cloud/learn/nosql-databases>
2. <https://www.coursera.org/lecture/nosql-databases/introduction-to-nosql-VdRNp>
3. <https://www.geeksforgeeks.org/introduction-to-nosql/>
4. <https://www.javatpoint.com/nosql-databa>

## COMPREHENSIVE VIVA-VOCE

<b>Semester VI</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>Credits</b>
<b>Subject code: H5-CS</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>1</b>

### **Guidelines for Comprehensive Viva for B.E (CSE) HONOR Program**

There shall be a Comprehensive Viva – Voce in the Honors degree program. The Comprehensive viva- voce will be conducted by the committee consisting of Head of the department and two senior faculty members of the department. The Comprehensive Viva – voce is aimed at to assess the students’ understanding in various subjects he/she studies during the B.Tech. course of study. The Comprehensive Viva- Voce is valued for 60 marks by the committee. A student shall acquire 1 credit only when he/she secures 24 marks on aggregate out of 60 marks allocated. If he/she fails to get 24 marks out of 60, he/she has to re-register for the comprehensive viva-voce as and when it is offered.

**Preparation:** Review your coursework: Ensure you have a thorough understanding of the topics covered in your program courses.

**Study materials:** Familiarize yourself with textbooks, lecture notes, assignments, and any recommended supplementary resources.

**Practice:** Solve sample questions and problems related to the subjects to reinforce your understanding.

**Viva Format:** The Comprehensive Viva will be conducted by a panel of examiners consisting of faculty members. The panel may ask questions related to various topics covered in your BE program courses.

**Viva Content:** Expect questions that cover a range of topics such as algorithms, data structures, programming languages, software engineering principles, and computer architecture etc.

Questions may be theoretical, conceptual, or practical in nature. You may be asked to explain concepts, solve problems, write code on paper, or discuss real-world applications.

**Problem Solving:** Be prepared to demonstrate your problem-solving abilities, especially for questions related to algorithms and data structures. Explain your thought process when working through a problem. If asked to write code, focus on clarity, correctness, and efficiency. Be ready to explain your code, discuss any trade-offs, and justify your design decisions.

**Real-world Applications:** Be prepared to discuss how the concepts you've learned in your minor program are applied in real-world scenarios. Provide examples or case studies when relevant.

**Evaluation:** Your performance in the Comprehensive Viva will be evaluated based on your understanding of the subject matter, problem-solving skills, clarity of communication, and overall presentation.

### SWAYAM / NPTEL

Semester VI	L	T	P	Credits
Subject code: H6-CS	0	0	0	3

Student can choose any one of the two courses

#### 1. Reinforcement Learning

**Duration : 12 weeks**

Week 1: Introduction

Week 2: Bandit algorithms – UCB, PAC

Week 3: Bandit algorithms –Median Elimination, Policy Gradient

Week 4: Full RL & MDPs

Week 5: Bellman Optimality

Week 6: Dynamic Programming & TD Methods

Week 7: Eligibility Traces

Week 8: Function Approximation

Week 9: Least Squares Methods

Week 10: Fitted Q, DQN & Policy Gradient for Full RL

Week 11: Hierarchical RL

Week 12: POMDPs

#### TEXT BOOKS

1. Reinforcement Learning - An Introduction, S. Sutton and A. G. Barto, MIT Press. 1998.

## **2. Social Networks**

**Duration: 12 weeks**

Week 1: Introduction

Week 2: Handling Real-world Network Datasets

Week 3: Strength of Weak Ties

Week 4: Strong and Weak Relationships (Continued) & Homophily

Week 5: Homophily Continued and +Ve / -Ve Relationships

Week 6: Link Analysis

Week 7: Cascading Behaviour in Networks

Week 8: Link Analysis (Continued)

Week 9: Power Laws and Rich-Get-Richer Phenomena

Week 10: Power law (contd.) and Epidemics

Week 11: Small World Phenomenon

Week 12: Pseudocore (How to go viral on web)

### **TEXT BOOKS**

1. Networks, Crowds and Markets by David Easley and Jon Kleinberg, Cambridge University Press, 2010
2. Social and Economic Networks by Matthew O. Jackson, Princeton University Press, 2010.

## GO Lang

<b>Semester VI</b>		<b>L</b>	<b>T</b>	<b>P</b>	<b>Credits</b>
<b>Subject code -H6-CS</b>	<b>0</b>	<b>0</b>	<b>3</b>	<b>3</b>	

<b>Course Objectives:</b>	<b>Course Outcomes:</b>
<ul style="list-style-type: none"> <li>➤ Develop a strong foundational understanding of Go programming.</li> <li>➤ Gain expertise in applying Go to implement machine learning algorithms and natural language processing techniques</li> <li>➤ Extend Go programming skills in concurrency, parallelism, and optimization for data-intensive applications.</li> <li>➤ Apply knowledge to develop a project that integrates Go programming with data science and NLP principles.</li> </ul>	<ol style="list-style-type: none"> <li>1. Demonstrate a deep understanding of Go programming fundamentals</li> <li>2. Techniques to manipulate data.</li> <li>3. Implement machine learning algorithms and NLP techniques using Go.</li> <li>4. Ability to apply advanced Go programming concepts like concurrency and optimization techniques</li> <li>5. Effectively integrates Go programming with data science and NLP principles,</li> </ol>

### UNIT I

**Introduction to Data Science and Go Programming:** Overview of Data Science and its Applications, Introduction to the Go Programming Language, Setting Up the Development Environment, Basic Syntax and Data Structures in Go.

### UNIT II

**Data Preprocessing with Go:** Handling Missing Data, Outlier Detection and Treatment, Outlier Detection and Treatment, Data Normalization and Standardization, Measures of Data Similarity and Dissimilarity

**Data Manipulation and Analysis with Go:** Data Import and Export in Go, Data Cleaning and Transformation, Data Visualization with Go, Exploratory Data Analysis (EDA), Statistical Analysis in Go

### UNIT III

**Machine Learning in Go:** Introduction to Machine Learning, Supervised Learning with Go (Regression and Classification), Unsupervised Learning with Go (Clustering and Dimensionality Reduction)

Evaluation Metrics for Machine Learning Models, Implementing Machine Learning Algorithms in Go

### UNIT IV

**Natural Language Processing (NLP) with Go:** Introduction to NLP and Its Applications Tokenization and Text Preprocessing in Go, Sentiment Analysis and Text Classification, Named Entity Recognition (NER) with Go, Building NLP Applications with Go.

### UNIT V

**Case Studies:** Concurrency and Parallelism in Go, Optimization Techniques for Data

### **TEXT BOOKS**

1. The Go Programming Language, Alan A. A. Donovan and Brian W. Kernighan, I Edition, Pearson Education,
2. An Introduction to Programming in Go, Caleb Doxsey 2012,ISBN: 978-1478355823

### **REFERENCE BOOKS**

1. Data Mining -Concepts and Techniques, by Jiawei Han, MichelineKamber, Jian Pei III Edition, Morgan Kaufmann Publisher, 2012.
- 2.Natural Language Processing in Action: Understanding, analysing, and generating text with Python by Lane, Hobson Lane, Hannes Hapke, Cole Howard, Manning,2019.

## **VII SEMESTER**

## RESEARCH METHODOLOGY

<b>Semester VII</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>Credits</b>
<b>Subject code: H7-CS</b>	<b>2</b>	<b>0</b>	<b>0</b>	<b>2</b>

<b>Course Objectives:</b>	<b>Course Outcomes:</b>
<ul style="list-style-type: none"><li>➤ Gain an appreciation for the diverse areas encompassing modern computer science, data science, and cybersecurity</li><li>➤ Understand different approaches to research in computer science Be able to read and ask questions in a computer science research paper</li><li>➤ Present and answer questions about a technical paper</li><li>➤ Understand how graduate education in computer science works</li></ul>	<ol style="list-style-type: none"><li>1. Understand the fundamentals of computer science research and its significance.</li><li>2. Identify and analyze research problems in computer science.</li><li>3. Evaluate and critique existing research papers and studies.</li><li>4. Collect and analyze research data.</li><li>5. Formulate research questions and hypotheses.</li></ol>

### **Week 1-2: Introduction to Computer Science Research**

- Overview of research in computer science
- Types of research (basic, applied, theoretical, empirical)
- The research process and methodology

### **Week 3-4: Literature Review**

- Conducting literature searches
- Critical reading and analysis of research papers
- Developing a research bibliography

### **Week 5-6: Research Problem Identification**

- Identifying research problems and gaps
- Formulating research questions and hypotheses

### **Week 7-8: Research Design and Methodology**

- Experimental design
- Surveys and case studies
- Ethical considerations in research

### **Week 9-10: Data Collection and Analysis**

- Data collection methods
- Data analysis techniques
- Statistical tools in research



## **Week 11-12: Research Presentation and Communication**

- Writing research papers
- Oral presentations
- Poster presentations

### **TEXTBOOKS**

1. Research Methods: A Practical Guide for the Social Sciences by W. Lawrence Neuman
2. How to Write and Publish a Scientific Paper by Robert A. Day and Barbara Gastel
3. The Craft of Research by Wayne C. Booth, Gregory G. Colomb, and Joseph M. Williams
4. The Elements of Style by William Strunk Jr. an

## PROJECT WORK

<b>Semester VII</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>Credits</b>
<b>Subject code: H8-CS</b>	<b>0</b>	<b>0</b>	<b>4</b>	<b>4</b>

The Viva-Voce shall be conducted by a committee consisting of HOD, Project Supervisor and an External Examiner nominated by the University. The Internal Evaluation shall be made by the departmental committee, on the basis of two seminars given by each student on the topic of his/her project.

### **Project Proposal:**

- Begin by developing a clear and well-defined project proposal. This should include a project title, objectives, scope, and a brief overview of the problem or area of interest.
- Specify the technologies, tools, and programming languages that will be used in the project.

### **2. Project Advisor:**

- Assign a faculty member as a project advisor to guide and mentor the student throughout the project.

### **3. Project Selection:**

- Choose a project that aligns with the program's objectives and your own interests. The project should be challenging and relevant to the field of computer science and engineering.
- Consider projects that involve software development, algorithm design, database management, data analysis, or other relevant areas.

### **4. Research and Literature Review:**

- Conduct a thorough literature review to understand existing solutions and research related to your project.
- Identify gaps in the current knowledge and explain how your project will contribute to addressing these gaps.

### **5. Implementation:**

- Begin the implementation phase by writing code, developing algorithms, or creating software as per your project's requirements.
- Ensure that your code adheres to coding standards and best practices.

#### **6. Testing and Debugging:**

- Rigorously test your project to identify and resolve bugs and errors.
  
- Perform unit testing, integration testing, and user acceptance testing as applicable.

#### **7. Documentation:**

- Maintain comprehensive documentation throughout the project. This includes code comments, user manuals, design documents, and technical reports.
- Properly cite and reference any external sources or libraries used in your project.

#### **8. Presentation and Demo:**

- Prepare a well-structured presentation and a live demonstration of your project's functionality.
- Highlight the problem statement, methodology, key features, and the impact of your project.

**11. Final Report:** - Submit a comprehensive final report that summarizes your project from start to finish. Include all documentation, code, and research findings.

**12. Presentation:** - Be prepared to present and defend your project in front of panel.

**13. Future Work and Impact:** - Discuss potential future work or enhancements that could be made to your project.

