

CS601 Machine Learning

THEORY:

Unit –I

Introduction to machine learning, scope and limitations, regression, probability, statistics and linear algebra for machine learning, convex optimization, data visualization, hypothesis function and testing, data distributions, data preprocessing, data augmentation, normalizing data sets, machine learning models, supervised and unsupervised learning.

Unit –II

Linearity vs non linearity, activation functions like sigmoid, ReLU, etc., weights and bias, loss function, gradient descent, multilayer network, backpropagation, weight initialization, training, testing, unstable gradient problem, auto encoders, batch normalization, dropout, L1 and L2 regularization, momentum, tuning hyper parameters,

Unit –III

Convolutional neural network, flattening, subsampling, padding, stride, convolution layer, pooling layer, loss layer, dance layer 1x1 convolution, inception network, input channels, transfer learning, one shot learning, dimension reductions, implementation of CNN like tensor flow, keras etc.

Unit –IV

Recurrent neural network, Long short-term memory, gated recurrent unit, translation, beam search and width, Bleu score, attention model, Reinforcement Learning, RL-framework, MDP, Bellman equations, Value Iteration and Policy Iteration, , Actor-critic model, Q-learning, SARSA

Unit –V

Support Vector Machines, Bayesian learning, application of machine learning in computer vision, speech processing, natural language processing etc, Case Study: ImageNet Competition

CS602 Computer Networks

THEORY:

Unit –I

Computer Network: Definitions, goals, components, Architecture, Classifications & Types. Layered Architecture: Protocol hierarchy, Design Issues, Interfaces and Services, Connection Oriented & Connectionless Services, Service primitives, Design issues & its functionality. ISO- OSI Reference Model: Principle, Model, Descriptions of various layers and its comparison with TCP/IP. Principles of physical layer: Media, Bandwidth, Data rate and Modulations

Unit-II

Data Link Layer: Need, Services Provided, Framing, Flow Control, Error control. Data Link Layer Protocol: Elementary & Sliding Window protocol: 1-bit, Go-Back-N, Selective Repeat, Hybrid ARQ. Protocol verification: Finite State Machine Models & Petri net models. ARP/RARP/GARP

Unit-III

MAC Sub layer: MAC Addressing, Binary Exponential Back-off (BEB) Algorithm, Distributed Random Access Schemes/Contention Schemes: for Data Services (ALOHA and Slotted- ALOHA), for Local-Area Networks (CSMA, CSMA/CD, CSMA/CA), Collision Free Protocols: Basic Bit Map, BRAP, Binary Count Down, MLMA Limited Contention Protocols: Adaptive Tree Walk, Performance Measuring Metrics. IEEE Standards 802 series & their variant.

Unit-IV

Network Layer: Need, Services Provided, Design issues, Routing algorithms: Least Cost Routing algorithm, Dijkstra's algorithm, Bellman-ford algorithm, Hierarchical Routing, Broadcast Routing, Multicast Routing. IP Addresses, Header format, Packet forwarding, Fragmentation and reassembly, ICMP, Comparative study of IPv4 & IPv6

Unit-V

Transport Layer: Design Issues, UDP: Header Format, Per-Segment Checksum, Carrying Unicast/Multicast Real-Time Traffic, TCP: Connection Management, Reliability of Data Transfers, TCP Flow Control, TCP Congestion Control, TCP Header Format, TCP Timer Management. Application Layer: WWW and HTTP, FTP, SSH, Email (SMTP, MIME, IMAP), DNS, Network Management (SNMP).

Departmental Elective - CS603 (C) Compiler Design

THEORY:

Unit-I Introduction to compiling & Lexical Analysis

Introduction of Compiler, Major data Structure in compiler, types of Compiler, Front-end and Back-end of compiler, Compiler structure: analysis-synthesis model of compilation, various phases of a compiler, Lexical analysis: Input buffering , Specification & Recognition of Tokens, Design of a Lexical Analyzer Generator, LEX.

Unit-II Syntax Analysis & Syntax Directed Translation

Syntax analysis: CFGs, Top down parsing, Brute force approach, recursive descent parsing, transformation on the grammars, predictive parsing, bottom up parsing, operator precedence parsing, LR parsers (SLR, LALR, LR), Parser generation. Syntax directed definitions: Construction of Syntax trees, Bottom up evaluation of S-attributed definition, L-attribute definition, Top down translation, Bottom Up evaluation of inherited attributes Recursive Evaluation, Analysis of Syntax directed definition.

Unit-III Type Checking & Run Time Environment

Type checking: type system, specification of simple type checker, equivalence of expression, types, type conversion, overloading of functions and operations, polymorphic functions. Run time Environment: storage organization, Storage allocation strategies, parameter passing, dynamic storage allocation , Symbol table, Error Detection & Recovery, Ad-Hoc and Systematic Methods.

Unit –IV Code Generation

Intermediate code generation: Declarations, Assignment statements, Boolean expressions, Case statements, Back patching, Procedure calls Code Generation: Issues in the design of code generator, Basic block and flow graphs, Register allocation and assignment, DAG representation of basic blocks, peephole optimization, generating code from DAG.

Unit –V Code Optimization

Introduction to Code optimization: sources of optimization of basic blocks, loops in flow graphs, dead code elimination, loop optimization, Introduction to global data flow analysis, Code Improving transformations ,Data flow analysis of structure flow graph Symbolic debugging of optimized code.

Open Elective - CS604 (B) Project Management

Course Learning Objectives:

Understand the different activities in software project development i.e, planning, design and management.

Course content:

1. Conventional Software Management.

Evolution of software economics. Improving software economics: reducing product size, software processes, team effectiveness, automation through software environments. Principles of modern software management.

2. Software Management Process

Framework,: Life cycle phases- inception, elaboration, construction and training phase. Artifacts of the process- the artifact sets, management artifacts, engineering artifacts, pragmatics artifacts. Model based software architectures. Workflows of the process. Checkpoints of the process.

3. Software Management Disciplines

Iterative process planning. Project organisations and responsibilities. Process automation. Project control And process instrumentation- core metrics, management indicators, life cycle expectations. Process discriminants.

Course Outcomes:

1. Understanding the evolution and improvement of software economics according to the basic parameters and transition to the modern software management.
2. Learning the objectives, activities and evaluation criteria of the various phases of the life cycle of software management process.
3. Gaining knowledge about the various artifacts, workflows and checkpoints of the software management process and exploring the design concept using model based architecture from technical and management perspective.
4. Develop an understanding of project planning, organization, responsibilities, automation and control of the processes to achieve the desirable results.