Unit – III
REINFORCEMENT LEARNING & NATURAL LANGUAGE PROCESSING

JNTUA SYLLABUS

Unit – III: **Reinforcement Learning**: Introduction, Passive Reinforcement Learning, Active Reinforcement Learning, Generalization in Reinforcement Learning, Policy Search, applications of RL

Natural Language Processing: Language Models, Text Classification, Information Retrieval, Information Extraction.

CONTENTS

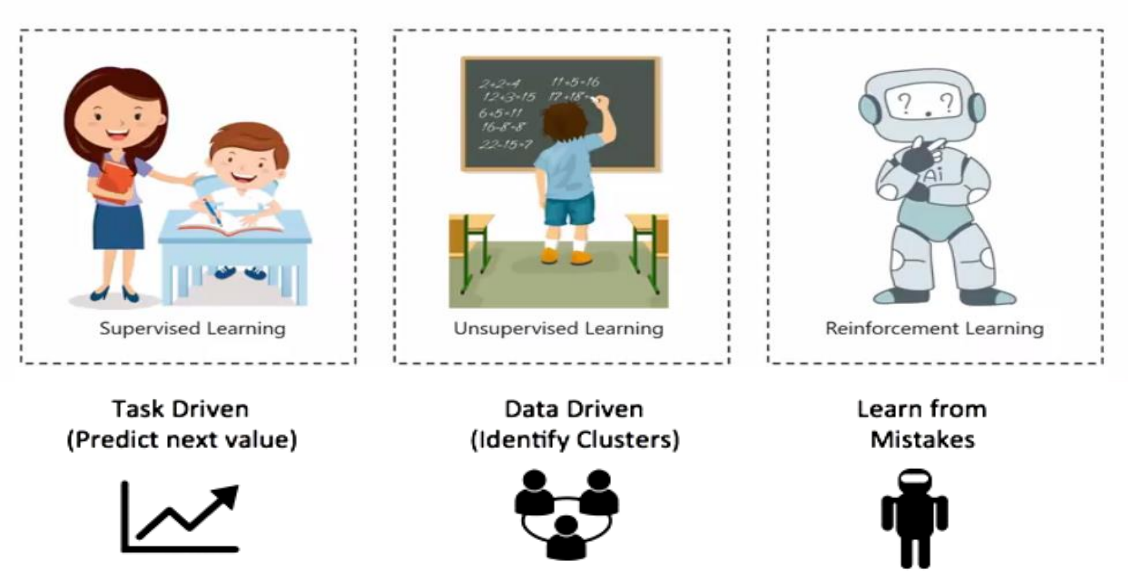
Unit –III
REINFORCEMENT LEARNING

3.1 MACHINE LEARNING

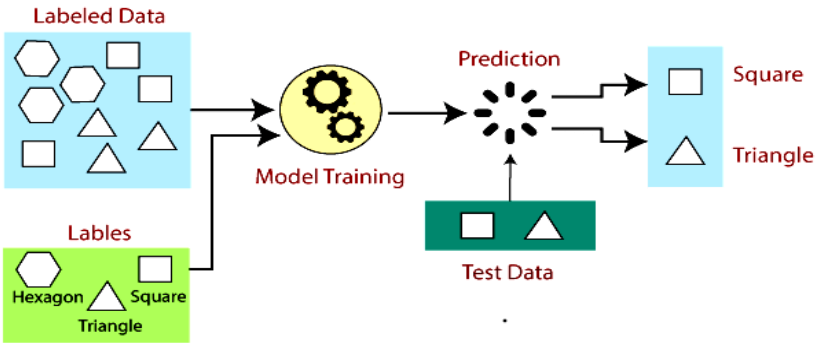
- Learning is any process by which a system improves performance from experience.
- Machine learning is concerned with computer programs that automatically improve their performance through experience.
- Machine learning allows the systems to make decisions automatically without any external support.
- These decisions are made when the machine is able to learn from the data and understand the underlying patterns that are contained within it.
- Then through pattern matching and further analysis, they return to outcome which can be a classification or a prediction.
- The difference between normal computer software and machine learning is that a human developer has not given codes that instruct the system how to react to situation; instead it is being trained by a large number of data.
- **Definition**: Machine learning is an application of Artificial Intelligence that involves algorithms and data that automatically analyze and make decision by itself without human intervention.

3.2 TYPES OF MACHINE LEARNING

- Machine Learning programs are classified into 3 types
    1. Supervised learning
    2. Unsupervised learning
    3. Reinforcement Learning



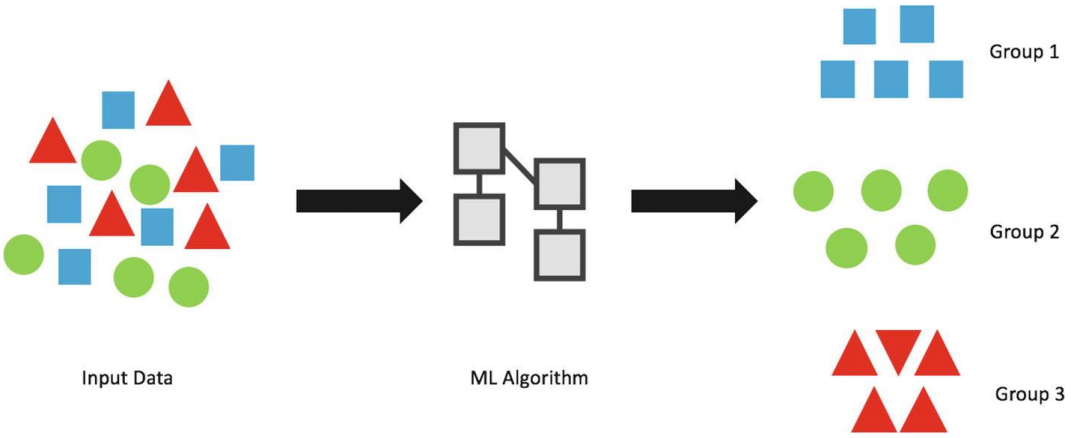
3.2.1 Supervised Learning (SL)



- Supervised learning is a technique where the program is given labeled input data and the expected output data.
- It gets the data from training data containing sets of examples.
- In supervised learning, models are trained using labeled data.
- Models need to find the mapping function to map the input variable (X) with the output variable (Y). i.e., **Y= f(X)**
- Supervised learning happens in the presence of a supervisor just like learning performed by a small child with the help of his teacher.
- A child trained to recognize fruits, colors, numbers under the supervision of a teacher is the best example of supervised learning.
- In this method, every step of the child is checked by the teacher and the child learns from the output that he has to produce.
- **Facial Recognition** is one of the most popular applications of Supervised Learning and more specifically – **Artificial Neural Networks**.
- ANN is used for identifying the faces of people. These models are able to draw features from the image through various filters. Finally, if there is a high similarity score between the input image and these image in the data base, a positive match is provided.
- Types of supervised learning algorithms.

  Classification : They notify the class of the data it is presented with. The classification algorithm separates the data into classes.
  Example - Is it the image of the sun or the moon?
  Classification of spam and non- spam mails.
  Regression : They expect the product to produce a numerical value.
  Example - The linear regression algorithm that is applied, predicts the cost of the house based on many parameters such as location, nearby airport, size of the house, etc.
  Weight of the person based on height, Salary of a person based on experience, Mileage of a car based on the speed and available fuel.
  Humidity of a location based on temperature, Price of the product over a period of time.

3.2.2 Unsupervised Learning (USL)



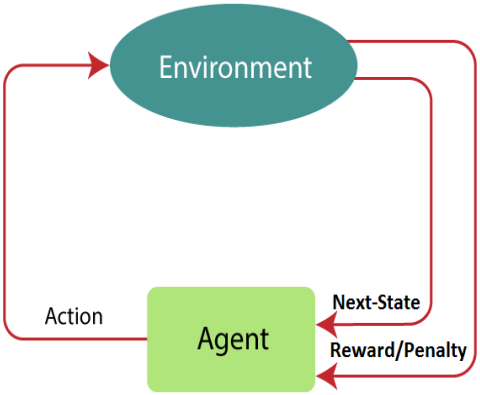
- Unsupervised learning happens without the help of a supervisor just like a fish learns to swim by itself.
- Unsupervised learning uses machine learning algorithms to analyze and cluster unlabeled data sets.
- These algorithms discover hidden patterns in data without the need for human intervention.
- It is an independent learning process.
- The goal of unsupervised learning is to find the structure and patterns from the input data.
- In this model, as there is no output mapped with the input, the target values are unknown/ unlabeled.
- The system needs to learn by itself from the data input to it and detect the hidden patterns.
- Types of unsupervised algorithm
  - ✓ **Clustering Algorithm**: This is a data mining technique for grouping unlabeled data based on their similarities or differences. The methods of finding the similarities between data items such as the same shape, size, color, price, etc. and grouping them to form a cluster for cluster analysis.
  - ✓ **Association Rule Mining**: This method uses different rules to find relationships between variables in a given dataset. These methods are frequently used for "market basket analysis" and recommendation engines, along the lines of "Customers Who Bought This Item Also Bought" recommendations.

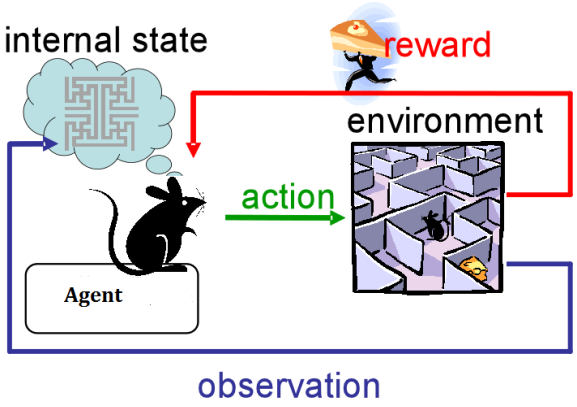
    Example – While buying products online, if bread is put in the cart, then it suggests buying butter, jam, cheese, etc.

    While purchasing shampoo it suggests hair conditioner, hair oil etc.,

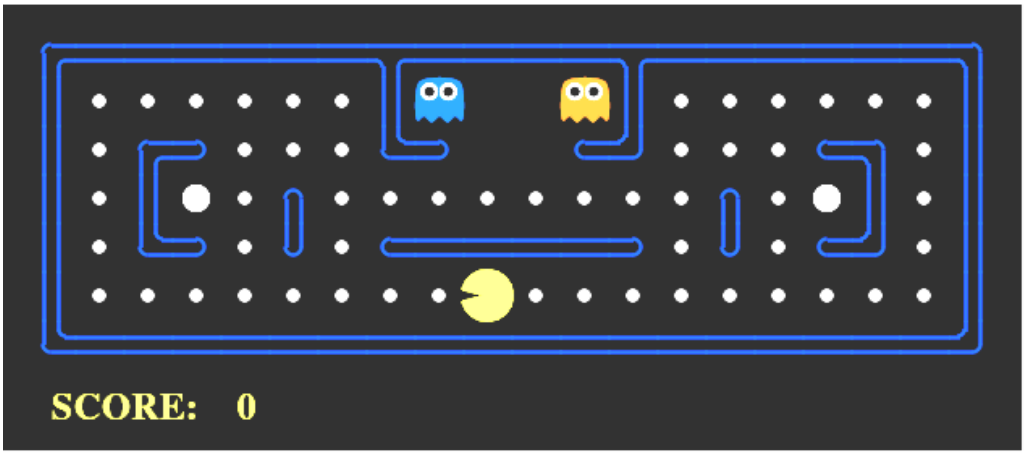
    While purchasing a laptop it may suggest mouse laptop bag, antivirus software plans, warranty plans etc.,

3.2.3 Reinforcement Learning (RL)





- Reinforcement Learning is a feedback-based Machine learning technique in which a software agent interacts with its environment by producing actions.
- For each good action, the agent gets positive feedback or reward, and for each bad action, the agent gets negative feedback or penalty.
- In Reinforcement Learning, the agent learns automatically using feedbacks without any labeled data.
- Since there is no labeled data, so the agent is bound to learn by its experience only.
- RL solves a specific type of problem where decision making is sequential, and the goal is long-term, such as **game-playing, robotics**, **self navigating vacuum cleaner e**tc.,
- The primary goal of an agent in RL is to improve the performance by getting the maximum positive rewards.
- The agent learns with the process of hit and trial, and based on the experience, it learns to perform the task in a better way.
- The agent continues doing these three things: **take action, change state/remain in the same state, and get feedback.**
- **Example : PacMan Game**



An RL problem can be best explained through games.
Let's take the game of **PacMan** where the goal of the agent (PacMan) is to eat the food in the grid while avoiding the ghosts on its way.
In this case, the grid world is the interactive environment for the agent where it acts.
Agent receives a reward for eating food and punishment if it gets killed by the ghost (loses the game). The states are the location of the agent in the grid world and the total cumulative reward is the agent winning the game.

3.2.3.1 Terms used in Reinforcement Learning

1. **Agent:** An entity which performs actions in an environment to gain some reward.
2. **Environment (e):** Physical world in which the agent operates. In RL, we assume the stochastic environment, which means it is random in nature.
3. **Action(a):** Actions are the moves taken by an agent within the environment.
4. **State(s):** Current situation of the agent.
5. **Reward(R):** Feedback from the environment.
6. **Policy(π):** Method or strategy to map agent's state to actions.
7. **Value(V):** Future reward that an agent would receive by taking an action in a particular state

3.2.3.2 Key Features of Reinforcement Learning

1. It is the art of optimal decision making process.
2. The agent is not instructed about the environment and what actions need to be taken.
3. It is based on the hit and trial process.
4. The agent takes the next action and changes states according to the feedback of the previous action.
5. The agent may get a delayed reward.
6. The environment is stochastic, and the agent needs to explore it to reach to get the maximum positive rewards.

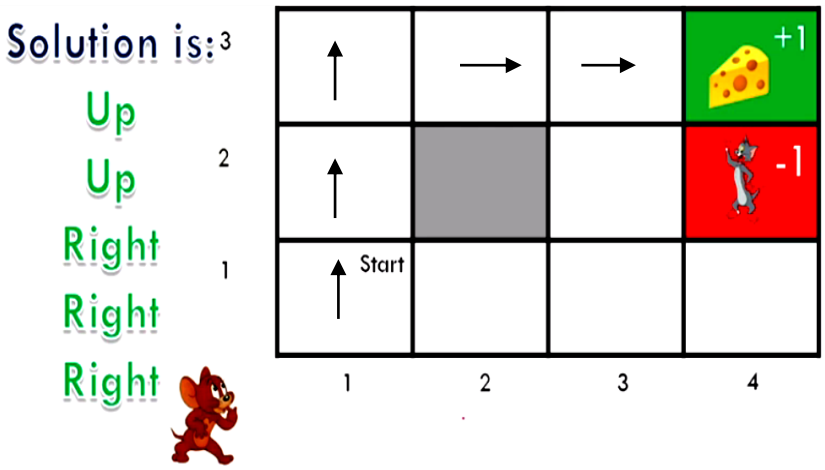
3.2.3.3 Reinforcement learning algorithms/Elements of RL

There are three approaches to implement a Reinforcement Learning algorithm.

1. **Value-Based algorithm**
   - The value function gives information about how good the situation and action are and how much reward an agent can expect.
   - A reward indicates the immediate signal for each good and bad action**,** whereas a value function specifies the good state and action for the future**.**
   - The value function depends on the reward as, without reward, there could be no value.
   - The goal of estimating values is to achieve more rewards.
   - In a value-based Reinforcement Learning method, we should try to maximize a value function **V(s)**.
   - In this method, the agent is expecting a long-term return of the current states under policy **π**.
2. **Policy-based algorithm**
   - A policy can be defined as a way how an agent behaves at a given time.
   - It maps the states of the environment to the actions taken on those states.
   - A policy is the core element of the RL as it alone can define the behavior of the agent.
   - In a policy-based RL method, every state helps to gain maximum reward in the future.
   - Two types of policy-based methods are:
     - ✓ Deterministic: For any state, the same action is produced by the policy **π**.
     - ✓ Stochastic: Every action has a certain probability.
3. **Model-Based algorithm**
   - Model predicts what the environment will do next?
   - In this method, a virtual model will be created for each environment.
   - This model mimics the behavior of the environment.
   - The agent learns to perform in that specific environment.
   - If a state and an action are given, then a model can predict the next state and reward.

3.3 PASSIVE REINFORCEMENT LEARNING

- In passive RL, the agents' policy **π** is fixed; it only needs to know how good it is.
- The agent task is to learn the utilities of various states.
- This is learning a model of the fully observable environment.
- It is useful for us to learn as one step on the way to active reinforcement learning.
- General examples : Students listening lectures, reading books, seeing pictures, watching videos, seeing & hearing demonstrations etc.,
- **Example : 4x3 World of Tom & Jerry**
  Reaching to the block of Bread gives a Reward of +1
  Reaching the block of Tom gives a Penalty of -1



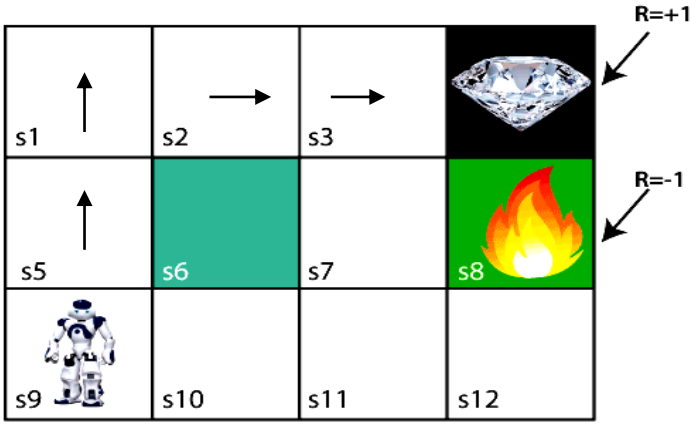
Solution : (1,1) (1,2) (1,3) (2,3) (3,3) (4,3)

- There are three methods of approach in Passive learning.
  - **Direct utility estimation**
  - **Adaptive dynamic programming(ADP)**
  - **Temporal difference learning(TDL)**

3.3.1. Direct Utility Estimation

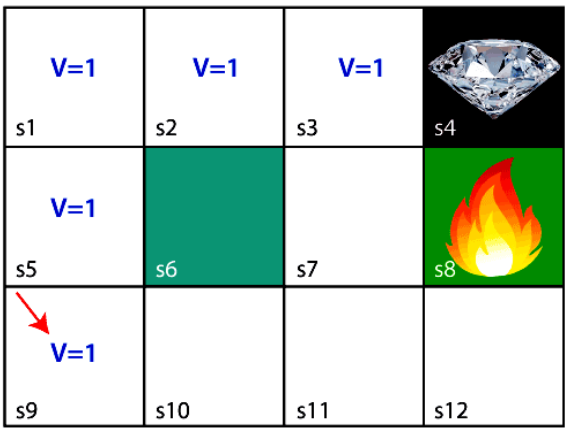
- In this method, the agent executes the policy, a bunch of times.
- The agent executes a sequence of trials or runs (sequences of states-actions transitions that continue until the agent reaches the terminal state).
- Each trial gives a sample value and the agent estimates the utility for each state based on the samples values.
- The main drawback is that this method makes a wrong assumption that state utilities are independent while in reality they are dependent. Also, it is a slow process.

- **Example : 4x3 World Maze Environment.**
  - Environment: It can be a room, maze, football ground, etc.
  - Agent**:** An intelligent agent such as AI robot.
  - Consider the below image.

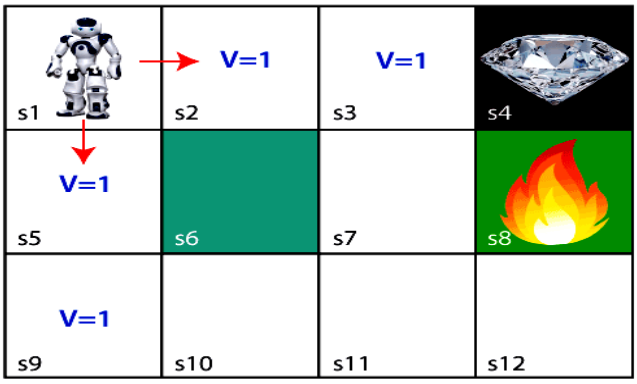


Lorem ipsum

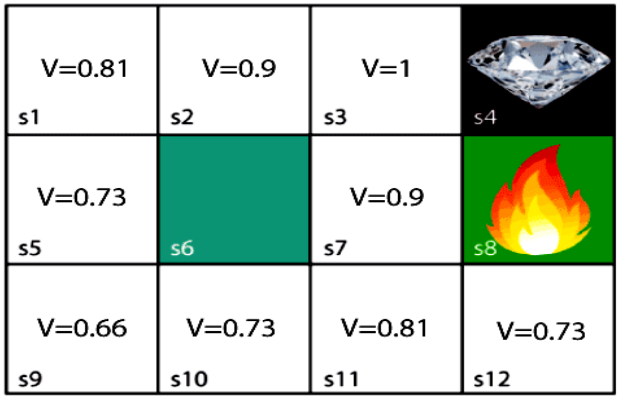
- In the above image, the agent is at S_9 block of the maze.
- S_6 block is a **wall**, S_8 a **fire pit**, and S_4 a **diamond block.**
- The agent cannot cross the S_6 block, as it is a solid wall.
- Reaching S4 block/ diamond leads to Reward +1
- Reaching the S8/fire leads to Penalty -1
- It can take four actions**:** move up, move down, move left, and move right.
- The agent can take any path to reach to the final point, but he needs to make it in possible fewer steps. Suppose the agent considers the path S9-S5-S1-S2-S3, so he will get the +1-reward point.
- The agent will try to remember the preceding steps that it has taken to reach the final step. To memorize the steps, it assigns 1 value to each previous step.
- Consider the below step.



- Now, the agent has successfully stored the previous steps assigning the 1 value to each previous block.
- But what will the agent do if he starts moving from the block, which has 1 value block on both sides?
- Consider the below diagram.



- It will be a difficult condition for the agent whether he should go up or down as each block has the same value.
- So, the above approach is not suitable for the agent to reach the destination.
- Hence to solve the problem Bellman equation can be used to find the value at each stage.
- Consider the below image with complete block wise values.



- As per the above image, the agent can decide the path of reaching the reward.
- If the agent is at S_5 block, it will move to S_1 block due to the higher value in S_1 block compared to S_9 block.
- Similarly, if the agent is at S_2 block, it will move to S_3 block due to the higher value in S_3 block compared to S_1 block.
- Using this approach, the agent will reach the diamond (reward point of +1) in S_4 block, completing the task.

3.3.2. Adaptive Dynamic Programming (ADP)

- ADP is a smarter method than Direct Utility Estimation.
- It runs trials to learn the model of the environment.
- It adopts the approach of modified policy iteration.
- It updates the utility estimates after each change.
- ADP is a model based approach and requires the transition model of the environment.

3.3.3. Temporal Difference Learning (TD)

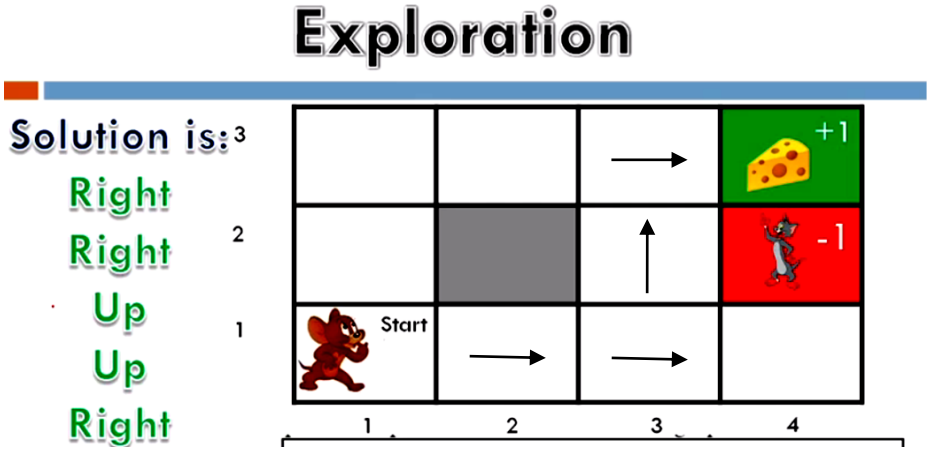
- A model-free approach is Temporal Difference Learning.
- TD learning does not require the agent to learn the transition model.
- The agent only updates states that are directly affected.
- It adjusts the values of the observed states.
- TD is slower but much simpler in terms of computation.

3.4 ACTIVE REINFORCEMENT LEARNING

- A passive learning agent has a fixed policy **π** that determines its behavior.
- An active RL agent must also learn what to do?
- Agent needs to learn a complete model with outcome probabilities for all actions.
- Agent has a choice of actions.
- Agent can extract optimal action by one-step look ahead to maximize the expected utility.
- The main concept is **exploration with experience**: an agent must experience as much as possible of its environment in order to learn how to behave in it.
- General Examples : Students participating in group discussion, Individual or group presentations, Role play, Doing experiments & projects, Doing case studies or simulations etc.,
- There are three methods of approach in Active learning.
  - **Adaptive dynamic programming with exploration function.**
  - **Q- learning.**
  - **SARSA (State-Action-Reward-State-Action)**

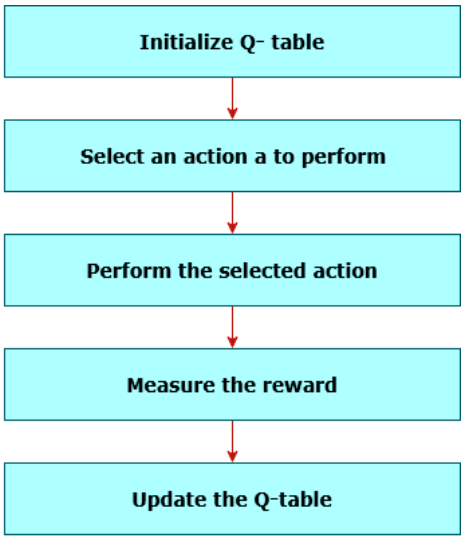
3.4.1 Adaptive Dynamic Programming with exploration function

- As the goal of an active agent is to learn an optimal policy, the agent needs to learn the expected utility of each state and update its policy.
- *The exploration function converts a passive agent into an active one.*
- Example : 4x3 World with exploration.
  Instead of reaching the goal in the path of (1,1) (1,2) (1,3) (2,3) (3,3) (4,3) it explores the other possibility like (1,1) (2,1) (3,1) (3,2) ((3,3) (4,3).



3.4.2. Q- Learning

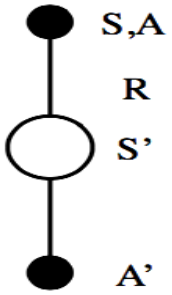
- The Q stands for quality in Q-learning, which means it specifies the quality of an action taken by the agent.
- Q learning is a value-based method of supplying information to inform which action an agent should take.
- This is basically a modification of the Temporal Difference algorithm.
- Q-learning is a TD learning method which does not require the agent to learn the transitional model, instead learns Q-value functions.
- The below flowchart explains the working of Q- learning:



3.4.3 State –Action- Reward –State- Action (SARSA)

- SARSA algorithm is a slight variation of the popular Q-Learning algorithm.
- The SARSA is named because it uses the quintuple **Q(s, a, r, s', a').**

  Where s: original state
  a: original action
  r: reward observed while following the states
  s' and a': New state, action pair.
- SARSA is an algorithm where, in the current state (S), an action (A) is taken and the agent gets a reward (R) and ends up in next state (S') and takes action (A') in S'. Therefore, quintuple (S, A, R, S', A') stands for the acronym **SARSA**.



- It is called an on-policy algorithm because it updates the policy based on actions taken.

3.5 GENERALIZATION IN REINFORCEMENT LEARNING

- Generalization refers to the model's ability to adapt properly to new, previously unseen data, drawn from the same distribution as the one used to create the model.
- A model trains on some data set of **N** samples and generates a label for each.
- Based on the empirical formulae and generated labels match an **empirical error** is calculated with **training data**.
- The model is then tested on a previously unseen data or **future data**, and an **expected error** is calculated based on its performance.
- **Generalization in RL** is that as **N** sample size increases, the difference between empirical error and the expected error goes to zero.
- **Training data :{Xi,Yi}**
  examples that we used to train our predictor
  e.g. all emails that are labeled spam/ham (not spam)
- **Future data or Testing data : {Xi,?}**
  examples that our classifier has never seen before.
  e.g. emails that will arrive tomorrow.

3.6 POLICY SEARCH

- In Reinforcement Learning, **Policy** :
  - Defines the agent's behavior at a given time.
  - A Method or strategy to map agent's state to actions.
- **Policy search** is a subfield in Reinforcement Learning (RL) which focuses on finding good parameters for a given policy.
- Policy-search methods operate directly on a representation of the policy, attempting to improve it based on observed performance.
- The variation in the performance in a stochastic domain is a serious problem; for simulated domains this can be overcome by fixing the randomness in advance.
- In Policy search, experience data will be provided.
- Model based RL learns the model with Adaptive Dynamic Programming and establishes a policy.
- Model free RL learns the value function and establishes a policy.
- Policy search will optimize the policy from the experience data and choosing the best one from model based RL and model free RL.

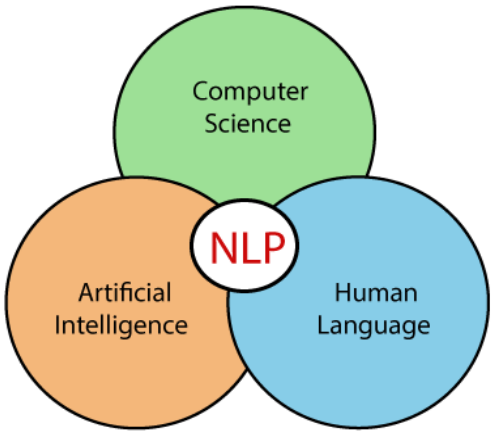
3.7 APPLICATIONS OF REINFORCEMENT LEARNING

1. Gaming applications
2. Robotics
3. Self-driving cars
4. Aircraft control
5. Industry automation
6. Marketing and advertising
7. Trading and finance.
8. Healthcare sector
9. Business strategy planning
10. Natural Language Processing

NATURAL LANGUAGE PROCESSING

3.8 NATURAL LANGUAGE PROCESSING (NLP)

- NLP is a branch of computer science, artificial intelligence and linguistics.
- NLP describes the interaction between human language and computer.
- NLP is the application of computational techniques to analyze large amount of natural language data and speech.
- NLP is concerned with giving computers, the ability to understand text and spoken words.
- NLP system accepts the inputs in the form of speech and text.
- Most of the NLP techniques use various supervised and unsupervised machine learning algorithms for extracting valuable insights from the human language.

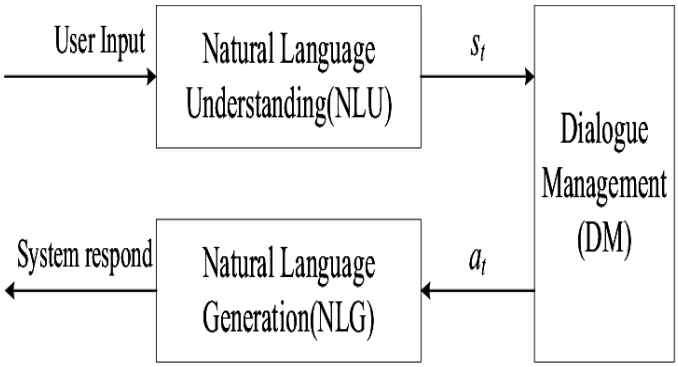


- A few examples of NLP that people use everyday are : Spell check, Search Autocomplete, Related keywords on search engines, Voice text messaging, Spam filters, Apple's Siri, Amazon's Alexa, Google Assistant, Advertising matching etc.,
- Natural Language Vs Computer Language

Natural Language	Computer Language
Natural language has a very large vocabulary.	Computer language has a very limited vocabulary.
E.g: English, Chinese, Telugu etc.,	E.g: C,C++,Java ,Python etc.,
Natural language is easily understood by humans.	Computer language is easily understood by the machines.
Natural language is ambiguous in nature.	Computer language is unambiguous.

3.9 COMPONENTS OF NLP

- The two main components of NLP are
  - Natural Language Understanding **(NLU)**
  - Natural Language Generation **(NLG)**



1. NLU –What users say?

- NLU is the process of taking some spoken/typed sentence and working out what it means?
- NLU deals with mapping the human input to natural language.
- NLU analyzes different aspects of the language.
- NLU helps the machine to understand and analyse human language by extracting the ***metadata** from content such as concepts, keywords, emotion, relations, and semantic roles.

  ***Metadata**–A set of data that describes & gives information about other data. Metadata of a music file may include artist's name, lyricist, music director, choreographer, year it was released etc.,
- NLU is mainly used for Business applications to understand the customer's problem in both spoken and written language.

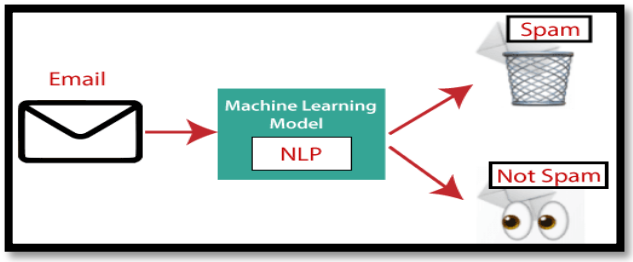
2. NLG – How to talk to users?

- NLG is the process of writing or generating language.
- NLG deals with structured data.
- NLG produces meaningful phrases and sentences.
- NLG acts as a translator that converts the computerized data into natural language representation.
- NLG should intelligent & conversational.

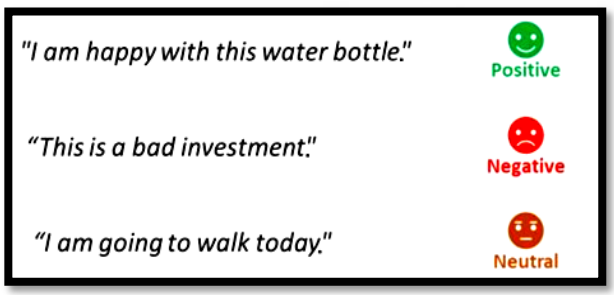
3.10 APPLICATIONS OF NLP/NLP USE CASES

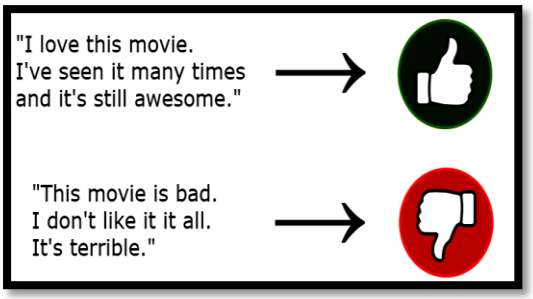
- NLP is the driving force behind machine intelligence in many modern real-world applications. Here are a few examples:

1. **Question Answering**
   Question Answering focuses on building systems that automatically answer the questions asked by humans in a natural language.
2. **Spam Detection**
   Spam detection is used to detect unwanted e-mails getting to a user's inbox.

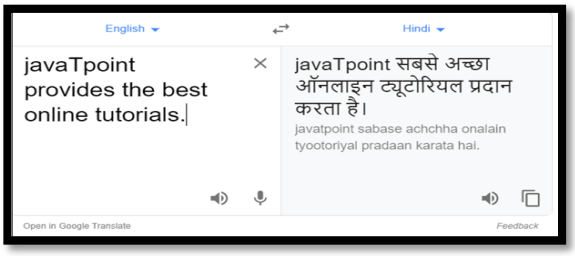


3. **Social Media Sentiment Analysis**
   Sentiment Analysis is also known as **opinion mining**. It is used on the web to analyse the attitude, behaviour, and emotional state of the sender. This application is implemented through a combination of NLP and statistics by assigning the values to the text (positive, negative, or natural), identify the mood of the context (happy, sad, angry, etc.)





4. **Machine Translation**
   Machine translation is used to translate text or speech from one natural language to another natural language.

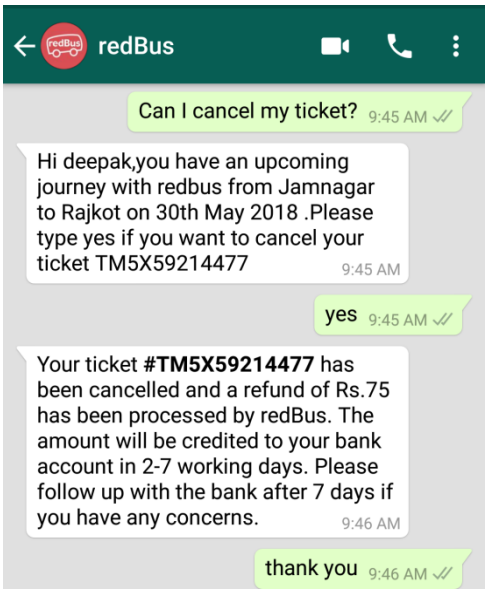


5. **Spelling correction**

   Microsoft Corporation provides word processor software like MS-word, PowerPoint for the spelling correction.

6. **Chatbot**

   Implementing the Chatbot is one of the important applications of NLP. It is used by many companies to provide the customer's chat services.



7. **Speech Recognition**

   Speech recognition is used for converting spoken words into text. It is used in applications, such as mobile, home automation, dictating to Microsoft Word, voice biometrics, voice user interface, and so on.

8. **Information extraction**

   It is used for extracting structured information from unstructured or semi-structured machine-readable documents.

9. **Virtual agents**

   Virtual agents such as Apple's Siri and Amazon's Alexa use speech recognition to recognize patterns in voice commands and natural language generation to respond with appropriate action or helpful comments.

3.11 LANGUAGE MODELS

- A language model in NLP computes the probability of a sentence (sequence of words) or the probability of a next word in a sequence.
- Probability of a sentence can be expressed as

$$\mathbf{P(W) = P(w_1, w_2, w_3, \ldots\ldots\ldots\ldots, w_n)}$$

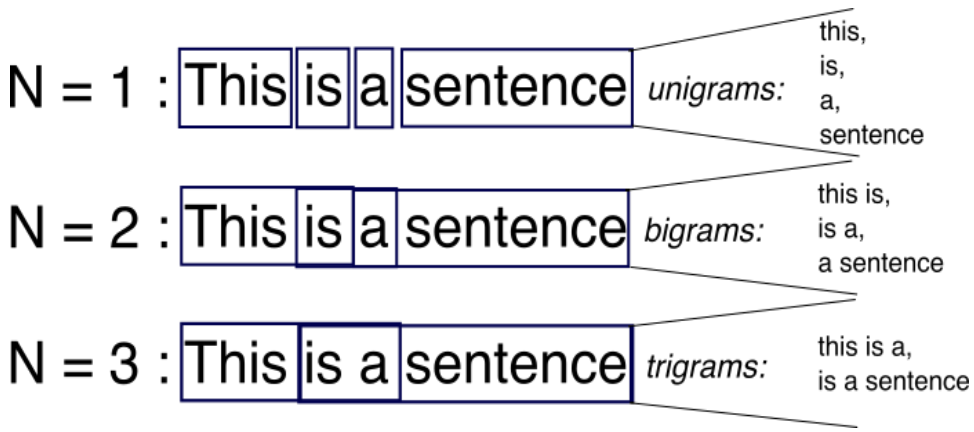
- Probability of the next word can be expressed as

$$\mathbf{P(w_5 | w_1, w_2, w_3, w_4)}$$

- Language models are used in speech recognition, machine translation, part-of-speech tagging, parsing, Optical Character Recognition, handwriting recognition, information retrieval etc.,.
- These language models power all the popular NLP applications like Google Assistant, Apple's Siri, Amazon's Alexa, etc.
- A **language** can be defined as a set of strings; "print(2 + 2)" is a legal program in the language Python, whereas "2)+(2 print" is not.
- Natural languages are difficult to deal with because they are very large, and constantly changing.

N-Gram Language Model

- An N-gram is a sequence of N words from a given corpus.
- This model estimates the probability of each word given prior context.
- A sequence of written symbols of length **n** is called an **n-gram**, with special case **"unigram" for 1-gram,**
  "bigram" for 2-gram, and
  "trigram" for 3-gram.
- This model is the probability distribution of **n-letter** sequences and hence is called an **n-gram model.**

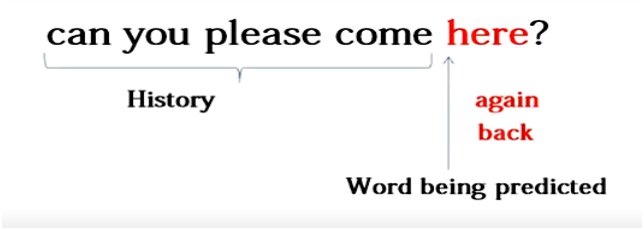


- **Example** : Consider the following sentence:

  "I love reading blogs about artificial intelligence."

- A 1-gram (or unigram) is a one-word sequence. For the above sentence, the unigrams would simply be: "I", "love", "reading", "blogs", "about", "artificial", "intelligence".
- A 2-gram (or bigram) is a two-word sequence of words, like "I love", "love reading", or "artificial intelligence".
- A 3-gram (or trigram) is a three-word sequence of words like "I love reading", "about artificial intelligence".

How do N-gram Language Models work?



- An N-gram language model predicts the probability of a given N-gram within any sequence of words in the language.
- A good N-gram model, can predict **p(w | H)** – what is the probability of seeing the **word *w*** given some **history H** (Previous words).
- We must estimate this probability to construct an N-gram model**.**
- We compute this probability in two steps:
  - Apply the chain rule of probability
  - We then apply a very strong simplification assumption to compute in an easy manner
- The chain rule of probability is:

  $$P(w_1.....w_n) = P(w_1) \cdot P(w_2 \mid w_1) \cdot P(w_3 \mid w_1\, w_2) \cdot \ ..... \ P(w_n \mid w_1...w_{n-1})$$

- **What is Markov chain rule?**

  It tells us how to compute the joint probability of a sequence by using the conditional probability of a word given previous words.
- But we do not have access to these conditional probabilities with complex conditions of up to n-1 words. So how do we proceed?
- This is where we introduce a simplification assumption. We can assume for all conditions, that:

  $$P(w_k \mid w_1...w_{k-1}) = P(w_k \mid w_{k-1})$$

- Here, we **approximate** the history (the context) of the word w_k by looking only at the last word of the context.
- This assumption is called the **Markov assumption**.

All the language models are classified as follows.

1. **N –Gram Model**
   - N-Gram: A sequence (of some unit- words, characters) of length n.
   - Unigram, Bigram and Triagrams for n=1,2, and 3.
   - Probability distribution of n-unit sequences
   - Follows Markov chain of order n -1
   - The probability of a unit depends only on some of the immediately preceding units.
2. **N-Gram character Models**
   - $P(C_{1:N})$ is the probability of a sequence of N *characters* C_1 through C_N
   - Typically corpus-based (uses a body of text)
   - In one Web collection P("the") = .03 and P("zgq") = .000000000002
   - It can handle unseen words.
   - Application: language identification like either English or German etc.,
   - Application : named- entity recognition(person ,location etc.,)
3. **Smoothing N-Gram Models**
   - What do we do about zero or low counts in a training corpus?
   - We will adjust the language model so that, sequences with count zero are assigned a small non-zero probability.
   - The process of adjusting the probability of low-frequency counts is called **smoothing**.
4. **Model Evaluation**
   - Just like Machine Learning, it cross-validates with training data/test data.
   - It splits the corpus into a training corpus and a validation corpus.
   - Application: language identification, Confused or puzzled words.
5. **N-Gram Word Models**
   - Turning N gram models over words rather than characters.
   - There are only about 100 characters in most languages.
   - With word models, we have at least tens of thousands of symbols, and sometimes millions.
   - The wide range is because it is not clear what constitutes a word.
   - In English a sequence of letters surrounded by spaces is a word, but in some languages, like Chinese, words are not separated by spaces.
   - "Word" needs to be defined precisely.
   - Word N gram models need to deal with **out of vocabulary** words.
   - This can be done by adding just one new word to the vocabulary: <UNK>, standing for the unknown word.
   - Application : Common in speech recognition

3.12 TEXT CLASSIFICATION

- This is also known as categorization.
- Language identification and genre (group) classification, sentiment analysis (classifying a movie or product review as positive or negative), spam detection (classifying an email message as spam or not-spam/ham) are examples of text classification.
- **Spam Attributes information**
  - ✓ Claim your reward
  - ✓ You are eligible for a free trip
  - ✓ Urgent: Your mobile number was awarded
  - ✓ Your account can be credited
  - ✓ free talk time offer on your phone number
  - ✓ Amazon free gift
  - ✓ Chance to win a SONY TV
  - ✓ Start earning salary
  - ✓ All medications at free of cost
  - ✓ Sta.rt earn*ing the salary yo,u d-eserve by o'btaining the prope,r crede'ntials!

- Word n-grams such as **"Claim your reward"** and **"You are eligible"** seem to be indicators of spam.
- Apparently the spammers can expect that the word trigram **"Claim your reward"** would be too indicative of spam.
- To overcome this situation, spammers may write as **"Cl,aim y*our re-ward".**
- A character model should detect this and we should create a full character n-gram model of spam and ham.
- We have two complementary ways of talking about classification.
  - In the language-modeling approach, we define one n-gram language model for **P(Message | spam)**by training on the spam folder, and
  - one model for **P(Message | ham)** by training on the inbox.

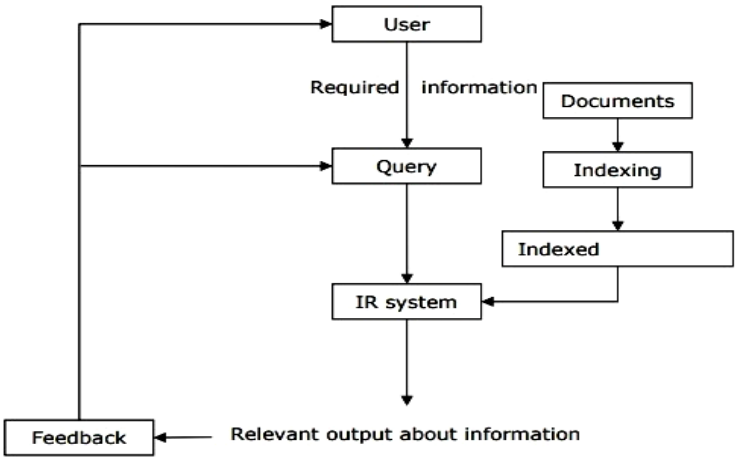
Classification by data compression: A compression algorithm takes a sequence of symbols, detects repeated patterns in it, and writes a description of the sequence that is more compact than the original.

- **Example:** the text "0.142857142857142857" might be compressed to "0.[142857]*3."
  Compression algorithms work by building dictionaries of subsequences of the text, and then referring to entries in the dictionary.
  The example here had only one dictionary entry, "142857."

.

3.13 INFORMATION RETRIEVAL (IR)

- IR is the process of accessing and retrieving the most appropriate information from text based on a particular query given by the user.
- IR is the task of finding documents that are relevant to a user's need for information.
- This is done with the help of context-based indexing or metadata.
- **Google, Yahoo, Bing** are the most famous example of IR.
- IR can be performed on the web, within emails, on a standalone machine (laptop) or in a company's knowledge base.
- A Web user can type a query such as [AI book] into a search engine and see a list of relevant pages.



- The step by step procedure of IR system is as follows:
  - Indexing the collection of documents.
  - Transforming the query in the same way as the document content is represented.
  - Comparing the description of each document with that of the query.
  - Listing the results in order of relevancy.
- IR systems consist of mainly two processes:
  i. **Indexing -** It is the process of selecting terms to represent a text.
  ii. **Matching -** It is the process of finding a measure of similarity between two text representations.
- IR System consists of the following.
  1. **IR Scoring Functions**
  2. **IR System Evaluation**
  3. **IR Refinements**
  4. **Page Rank Algorithm**
  5. **HITS (Hyperlink-Induced Topic Search) Algorithm**
  6. **Question Answering**

3.13.1 IR Scoring Functions

- Most IR systems use models, based on the statistics of word counts.
- A scoring function takes a document & a query and provides a numeric score.
- The most relevant documents have the highest scores.
- The relevance of a document is computed based on the following parameters

1. **Term Frequency (TF)**: This is simply the number of times a given term appears in that document.

   TF (i, j) = (count of i^{th} term in j^{th} document)/(total terms in j^{th} document)

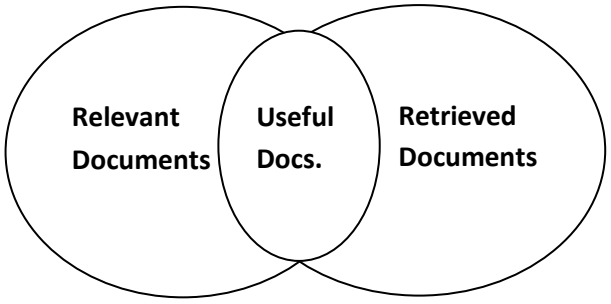
2. **Inverse Document Frequency (IDF)**: This is a measure of the general importance of the term.

   IDF (i) = (total no. of documents)/(no. of documents containing i^{th} term)

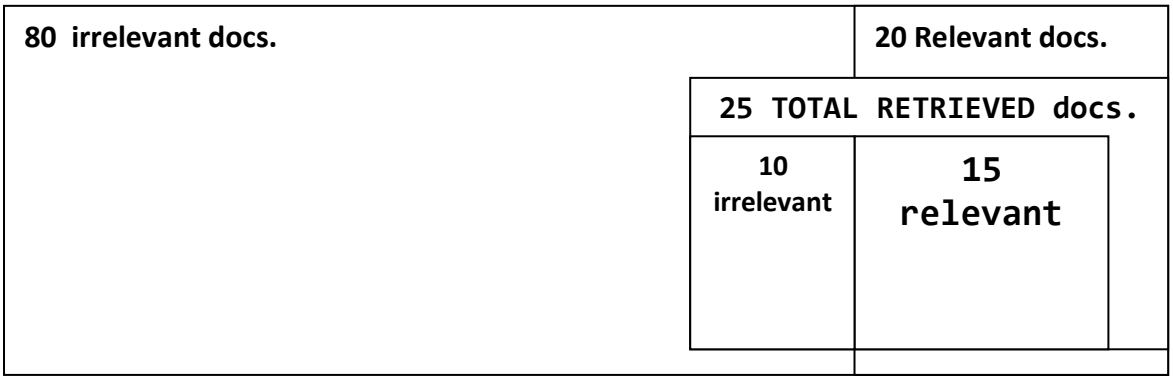
3. **TF-IDF Score (i, j) = TF * IDF**

3.13.2 IR System Evaluation

- The common measures for evaluating IR systems are: **Recall, Precision**



	retrieved	not retrieved
irrelevant	retrieved & irrelevant	Not retrieved & irrelevant
relevant	retrieved & relevant	not retrieved but relevant



Recall = No.of relevant documents retrieved/ Total no. of relevant documents
= 15/ 20 =0.75

Precision= No.of relevant documents retrieved /Total no.of retrieved documents
= 15 / 25 = 0.6

- In simple
  Recall – What percentage of the correct answers did the system get.
  Precision – What percentage of the system's answers were correct.
- Both Recall and Precession in IR system are measures of goodness that are tied to the concept of relevance.

3.13.3 IR Refinements

- Web search engines are continually updating their algorithms as they discover new approaches and as the Web grows and changes.
- In few cases, some words are correlated: "couch" is closely related to both "couches" and "sofa.". It should recognize these synonyms.
- A user who gives the query [Tim Couch] wants to see results about the football player, but not about sofa.
- Related words that are not synonyms also play an important role in ranking—terms like "leather", "wooden," or "modern" can serve to confirm that the document really is about "couch or sofa."
- As a final refinement, IR can be improved by considering **metadata.**

3.13.4 The PageRank algorithm

- PageRank (PR) is the first algorithm used by Google Search to rank web pages in their search engine results.
- PageRank is a way of measuring the importance of website pages.
- According to Google:
  "PageRank works by counting the number and quality of links to a page to determine a rough estimate of how important the website is. The underlying assumption is that more important websites are likely to receive more links from other websites."

3.13.5 HITS (Hyperlink-Induced Topic Search) Algorithm

- This algorithm is used in link analysis.
- It could discover and rank the webpages relevant for a particular search.
- The idea of this algorithm is that, an ideal website should link to other relevant sites and also being linked by other important sites.
- This is also known as "Hubs and Authorities".
  Each page in this set is considered an **authority** on the query to the degree
  A page is considered a **hub** to other authoritative pages.

3.13.6 Question Answering

- Question Answering focuses on building systems that automatically answer the questions asked by humans in a natural language.
  Example: **Google's Active Query Answering (AQA)** system reformulates the questions asked by the user**.**

If we ask the AQA bot the question – "What is the birth date of Gandhi " then the AQA bot would reformulate it into different questions like

"What is the birth year of Gandhi?"
"When was Gandhi born?" and
"When is Gandhi's birthday?"

- Google **AQA** system has integrated query based environment system.

3.14 INFORMATION EXTRACTION (IE)

- IE is a type of information retrieval.
- IE is used for extracting structured information from unstructured or semi-structured machine-readable documents.
- IE is the process of scanning text for relevant information to some interest.
- IE focus on relevant parts, ignores the rest.
- IE system extracts clear, factual information
  - Entities (person, organization etc.,)
  - Relations (position of a person in a company)
  - Events
  - Who did what to whom, when, where and why?
- IE can be applied to a wide range of textual sources
  - News paper articles
  - emails
  - Web pages
  - Reports, legal documents
  - Scientific articles
  - Newsgroup messages
  - Classified ads etc.,
- In IE, information may be single-slot or multi slot
  - Single slot - Single information: Microsoft CEO Satya Nadella.
  - Multi slot – Set of information : Rent – 1BHK Rs.7,000p.m , 2BHK Rs.15,000p.m
- **IR Vs IE**
  - IR retrieves relevant documents in a collection.
  - IE extracts relevant information from documents.
- IE uses NLP techniques/phases. They are
  1. Sentence Segmentation
  2. Word Tokenization
  3. Stemming
  4. Lemmatization
  5. Identify stop words
  6. POS tagging
  7. Named Entity Recognition

1. **Sentence Segmentation**

   It breaks the paragraph into separate sentences.

   Example: Consider the following paragraph -

   Independence Day is one of the important festivals for every Indian citizen. It is celebrated on the 15th of August each year. The day celebrates independence in the true sense.

   Sentence Segment produces the following result:

   a. "Independence Day is one of the important festivals for every Indian citizen."
   b. "It is celebrated on the 15th of August each year."
   c. "The day celebrates independence in the true sense."

2. **Word Tokenization**

   It is used to break the sentence into separate words or tokens.

   Example:

   "Tokenization is the main step in NLP"

   Word Tokenizer generates the following result:

   "Tokenization", "is", "the", "main", "step" "in"," NLP"

3. **Stemming**

   Stemming is used to normalize words into its base form or root form.

   Example: Affect, Affects, Affected, Affecting, Affection, Affectation. These words came from the root word of Affect.

   The big problem with stemming is that sometimes it produces the root word which may not have any meaning.

   For Example**,** intelligence, intelligent, and intelligently, all these words are originated with a single root word "intelligen." In English, the word "intelligen" do not have any meaning.

4. **Lemmatization**

   Lemmatization is quite similar to the Stemming.

   It is used to group different inflected forms of the word, called Lemma.

   The main difference between Stemming and lemmatization is that it produces the root word, which has a meaning.

   Example: The words like "Went", "Gone", "Going" has a root word of "Go", which has a meaning.

5. **Identifying Stop Words**

   In English, there are a lot of words that appear very frequently like "is", "and", "the", and "a". NLP pipelines will flag these words as stop words. Stop words might be filtered out before doing any statistical analysis.

   Example: He **is a** good boy.

6. **POS tagging**

   POS stands for Parts of Speech, which includes Noun, verb, adverb, and Adjective.

   It indicates that how a word functions with its meaning as well as grammatically within the sentences.

   A word has one or more parts of speech based on the context in which it is used.

   Example: "Google" something on the Internet.

   In the above example, Google is used as a verb, although it is a proper noun.

7. **Named Entity Recognition (NER)**

   NER is the process of detecting the named entity such as person name, movie name, organization name, location, quantity or value.

   Example: Google's CEO Sundar Pichai introduced the new Pixel3 at New York Central Mall.

   In the above statement, Sundar Pichai is a Person, New York is a location, Pixel3 is an item, Google and Central Mall are the organizations.

"Machine intelligence is the last invention that humanity will ever need to make."

- ***Nick Bostrom, Professor & Philosopher at University of Oxford known for his work on computational neuroscience, and artificial intelligence.***

Question Bank

Unit – III: REINFORCEMENT LEARNING & NATURAL LANGUAGE PROCESSING

2 Marks Questions

1. What is Machine learning?
2. Give the list of types of Machine learning.
3. Give the differences between supervised learning and unsupervised learning.
4. What is Reinforcement learning?
5. What is direct utility estimation?
6. Explain Adaptive Dynamic Programming
7. What is Temporal Difference Learning?
8. What is Q-learning?
9. Explain SARSA.
10. What is clustering algorithm?
11. What is association rule mining?
12. Explain the terms Action, Stage, Reward, and Policy.
13. Define NLP.
14. What are the components of NLP?
15. Discuss the importance and goals of NLP.
16. What is sentiment analysis in NLP?
17. Explain Machine translation.
18. What is a language model?
19. Explain classification by data compression.
20. Explain POS tagging with example.

Essay Questions

1. What is Reinforcement learning? Explain key features of Reinforcement learning.
2. Explain in detail about Passive Reinforcement learning.
3. What is Active Reinforcement learning? Explain.
4. Explain Generalization and Policy search in Reinforcement learning.
5. Explain the applications of Reinforcement Learning.
6. What are the components of NLP? Explain architecture of NLP.
7. What is NLP? List and explain the applications of NLP.
8. Explain Text classification.
9. What is Information Retrieval? Explain various systems in IR.
10. Explain Information Extraction.