Applied AI Unit 2

Applied AI Unit 2

Applied Artificial Intelligence

Unit 2

Probability Theory is a branch of mathematics that deals with the likelihood of events happening.

In simple words, probability answers the question:

What are the chances of something happening? or *How likely something is to happen.*

Why Learn Probability? Probability is used in:

- Weather forecasting ("60% chance of rain")
- Games and gambling (dice, cards, roulette)
- Medical diagnosis (probability of having a disease)
- Artificial Intelligence & Expert Systems (handling uncertain information)

Probability Theory

Term	Definition	Example
Experiment	An action with an uncertain result	Tossing a coin
Sample Space (S)	All possible outcomes	S = {Heads, Tails}
Event (E)	A subset of outcomes	E = {Heads}
Favorable Outcomes	Outcomes that match the event	1 (if we want Heads)

Formula of Probability

$$\text{Probability (P)} = \frac{\text{Number of Favorable Outcomes}}{\text{Total Number of Outcomes}}$$

Example 1: Tossing a Coin

- Sample Space: {Heads, Tails}
- Probability of getting Heads = P(Heads) = ½

Example 2: Rolling a Die

- Sample Space = {1, 2, 3, 4, 5, 6}

- Event: Getting a 3
- Favorable outcomes = 1
- Total outcomes = 6
- P(3) = 1/6

Example 3: Drawing a Card from a Deck

- Total cards in a deck = 52
- Event: Drawing a King
- Favorable outcomes = 4 (1 king in each suit)
- P(King)=4/52=1/13P

Types of Probability

1. Theoretical Probability

- Based on reasoning, not experiments.
- Example: Tossing a fair coin –

$$P(\text{Head}) = \frac{1}{2}$$

2. Experimental (Empirical) Probability

- Based on actual experiments or data.
- Example: You toss a coin 100 times and get 60 heads →

$$P(\text{Head}) = \frac{60}{100} = 0.6$$

What is Joint Probability?

- Joint probability is the probability of two events happening at the same time.
- $P(A \cap B)$=Probability of A and B occurring together

Example in General Terms:

Let's say:

- P(A) = Probability that a student studies AI = 0.6
- P(B) = Probability that a student studies Expert Systems = 0.5
- $P(A \cap B)$ = Probability that a student studies both = P(A) × P(B) = 0.6 × 0.5 = 0.3

This 0.3 is the joint probability.

Example in AI Context (Expert System):

Imagine an AI medical expert system analyzing diseases:

- A: The patient has fever → $P(A) = 0.8$
- B: The patient has cough → $P(B) = 0.6$
- $P(A \cap B)$ = Probability the patient has fever and cough together

If these are independent: $P(A \cap B) = P(A) \times P(B) = 0.8 \times 0.6 = 0.48$

Conditional Probability

- Conditional probability is the probability of event A occurring, given that event B has already happened.

Example in General Terms:

- $P(A \cap B)$ = Probability a student studies AI and scores high = 0.3
- $P(B)$ = Probability a student scores high = 0.6

$P(AI|High\ Score) = 0.3 / 0.6 = 0.5$

There's a 50% chance the student studies AI if we know they scored high.

Example in Expert Systems (Medical AI):

Let’s say:

- A: Patient has COVID
- B: Patient has fever
- $P(A \cap B) = 0.1$ (fever and COVID together)
- $P(B) = 0.4$ (fever observed)

$P(COVID|Fever) = 0.1 / 0.4=0.25$

So, there's a 25% chance a patient has COVID if they have a fever.

Used in expert systems to narrow down diagnoses.

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

Probability vs Fuzzy

Feature	Probability	Fuzzy Logic
Interpretation of 0.7	70% **chance** it will rain	"Rain is **moderately heavy**"
Truth Value	Event either happens or not	Truth can be **partial**
Focus	**Likelihood of events**	**Degree of truth or membership**
Example Statement	"There is a 0.6 probability that the patient has fever"	"The patient's fever is **0.6 high**"
Used For	Handling **randomness, statistical uncertainty**	Handling **imprecise concepts or linguistic terms**
Applications	Probability trees, Bayes' theorem, ML	Washing machines, expert systems, control systems

Bayes’ Theorem (1)

Bayes’ Theorem is a way to update probabilities based on new evidence.

- P(A): Prior probability of hypothesis A (before seeing data)
- P(B|A): Likelihood of observing B if A is true
- P(B): Total probability of evidence B
- P(A|B): Posterior probability (after seeing evidence B)

Example: Email Spam Detection (AI System)

- A = Email is spam
- B = Email contains the word “discount”

Assume:

- P(A) = Probability that any email is spam = 0.2
- P(B|A) = Probability "discount" appears in spam = 0.8
- P(B) = Probability "discount" appears in any email = 0.4

Apply Bayes’ Theorem:

P(Spam|Discount) = 0.8 × 0.2 / 0.4 = 0.4

✅ So, if an email contains "discount", there's a 40% chance it is spam.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

Bayes’ Theorem (2)

Example in Medical Expert System

- A = Patient has malaria
- B = Patient has chills

Assume:

- P(A) = 0.1 (prior probability of malaria)
- P(B|A) = 0.9 (chills are common in malaria)
- P(B) = 0.3 (probability a patient has chills in general)

P(Malaria | Chills) = 0.9 x 0.1 / 0.3 = 0.3

✔ The expert system can increase its belief in malaria diagnosis if chills are reported.

Term	Name	Meaning
P(A)	**Prior Probability**	Belief in hypothesis A **before** seeing the evidence
**P(B	A)**	**Likelihood**
P(B)	**Marginal Probability / Evidence**	Overall probability of observing B
**P(A	B)**	**Posterior Probability**

Bayes’ Theorem (3)

- Posterior Probability is the updated probability of a hypothesis after taking new evidence into account.
- It comes from Bayes' Theorem and helps refine our belief in something once we know more information.

Example:

Imagine you're guessing what's inside a box.

You think it’s probably a toy car (initial guess = prior probability).

Then you shake the box and hear metallic wheels rolling.

Now you're more confident it's a toy car — this updated belief is the posterior probability.

Term	Meaning
Hypothesis	A tentative conclusion based on limited data
Evidence	Observations or symptoms that support or contradict the hypothesis
Testing	Gathering more data to confirm or reject the hypothesis

Bayes' Theorem (4)

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

- **P(A)** → Prior probability (before evidence)
- **P(B|A)** → Likelihood (probability of the evidence given the hypothesis)
- **P(B)** → Total probability of the evidence
- **P(A|B)** → ⭐ **Posterior probability** (after seeing the evidence)

Real-World Example: Medical Diagnosis

Let's say:

- A = Patient has **malaria**
- B = Patient has **fever**

Known probabilities:

- **P(A)** = 0.1 (10% prior chance the patient has malaria)
- **P(B|A)** = 0.9 (90% chance a malaria patient has fever)
- **P(B)** = 0.3 (30% of all patients have fever)

📊 **Apply Bayes' Theorem:**

$$P(\text{Malaria}|\text{Fever}) = \frac{0.9 \cdot 0.1}{0.3} = \frac{0.09}{0.3} = 0.3$$

✅ **Posterior Probability = 0.3**
👉 So, after knowing the patient has fever, the chance they have malaria increases to **30%**.

Bayes Theorem in In Expert Systems or AI

In AI, **posterior probability** helps the system **adjust its decisions** based on **new observations**.

Scenario	Posterior Means
Email spam detection	Probability an email is spam **after seeing certain keywords**
Fraud detection	Probability a transaction is fraud **after seeing unusual location or amount**
Medical AI	Probability of a disease **after seeing lab results or symptoms**

Calculating Posterior Probability with Bayes' Theorem (1)

Objective:

Students will learn to update their beliefs (posterior probability) using new evidence by applying Bayes' Theorem in a real-world scenario.

Scenario: Medical Testing for a Disease

- Imagine a disease (let's call it Disease X) in a population. You are provided with the following data:

- Prevalence (Prior Probability): The probability that a random person has Disease X is 5% (P(Disease) = 0.05).
- Test Sensitivity: The test correctly identifies someone with Disease X 90% of the time (P(Positive Test | Disease) = 0.9).
- Test Specificity: The test correctly identifies someone without Disease X 80% of the time. Therefore, the false positive rate is 20% (P(Positive Test | No Disease) = 0.2).

Task:

Calculate the posterior probability—the probability that a person has Disease X given that they have tested positive.

Calculating Posterior Probability with Bayes' Theorem (2)

Step-by-Step Instructions:

Step 1: Define the Events

- Let D represent the event that a person has Disease X.
- Let T represent the event that the test result is positive.

Given:

- P(D) = 0.05 (Prior probability of Disease X)
- P(T | D) = 0.9 (Test sensitivity: probability of a positive test if the person has the disease)
- P(T | ¬D) = 0.2 (False positive rate: probability of a positive test if the person does not have the disease)

Calculating Posterior Probability with Bayes' Theorem (3)

Step 2: Find the Marginal Probability of a Positive Test (P(T))

Use the law of total probability:

$$P(T) = P(T|D) \cdot P(D) + P(T|\neg D) \cdot P(\neg D)$$

Where:

- $P(\neg D) = 1 - P(D) = 0.95$

Plug in the numbers:

$$P(T) = (0.9 \times 0.05) + (0.2 \times 0.95)$$

$$P(T) = 0.045 + 0.19 = 0.235$$

Step 3: Apply Bayes' Theorem

Bayes' Theorem is given by:

$$P(D|T) = \frac{P(T|D) \cdot P(D)}{P(T)}$$

Plug in the values:

$$P(D|T) = \frac{0.9 \times 0.05}{0.235} = \frac{0.045}{0.235}$$

Calculating Posterior Probability with Bayes' Theorem (4)

Step 3: Apply Bayes' Theorem

Bayes' Theorem is given by:

$$P(D|T) = \frac{P(T|D) \cdot P(D)}{P(T)}$$

Plug in the values:

$$P(D|T) = \frac{0.9 \times 0.05}{0.235} = \frac{0.045}{0.235}$$

Step 4: Compute the Posterior Probability

Calculate the division:

$$P(D|T) \approx 0.1915 \text{ or } 19.15\%$$

What is the Law of Total Probability?

- The Law of Total Probability helps us find the overall probability of an event by breaking it down into parts — specifically when the outcome depends on different conditions or scenarios.
- If an event can happen in multiple ways, the total probability of that event is the sum of the probabilities of it happening in each way.
- Think of the law as saying: "To find the total chance of something happening, look at all the different situations it could happen in, and add them up, weighted by how often each situation occurs."

Let $B_1, B_2, \ldots, B_n$ be a **partition** of the sample space (i.e., all the possible, non-overlapping ways an event can occur), and A is an event of interest.

$$P(A) = P(A|B_1) \cdot P(B_1) + P(A|B_2) \cdot P(B_2) + \cdots + P(A|B_n) \cdot P(B_n)$$

Real-Life Analogy: Pizza Delivery (1)

Scenario:

You're tracking how often a pizza is delivered on time (Event A) depending on the delivery person (B_1, B_2, B_3).

- $P(A \mid B_1) = 90\%$ (if John delivers it, it's on time 90% of the time)

This is the conditional probability:

"Given that John delivered the pizza, what's the probability it arrived on time?"

- $P(A \mid B_2) = 70\%$ (if Mike delivers it, it's on time 70% of the time)
- $P(A \mid B_3) = 50\%$ (if Anna delivers it, it's on time 50% of the time)

Real-Life Analogy: Pizza Delivery (2)

Let’s say:

- John delivers 50% of the time → $P(B_1) = 0.5$

This is the probability that John was the delivery person at all.

“Out of all pizza deliveries, how often is John the one delivering?”

- Mike delivers 30% of the time → $P(B_2) = 0.3$
- Anna delivers 20% of the time → $P(B_3) = 0.2$
- "Probability of a successful outcome = How good each person is × How often that person shows up"

So we combine them in the Law of Total Probability:

$$P(\text{Pizza on Time}) = P(\text{On Time}|\text{John}) \times P(\text{John Delivers}) + \ldots$$
$$= 0.9 \times 0.5 + (\text{others...})$$

Real-Life Analogy: Pizza Delivery (3)

💡 **Question:**

What is the **overall probability** that a pizza is delivered on time?

✏️ **Apply the Law of Total Probability:**

$$P(A) = (0.9 \times 0.5) + (0.7 \times 0.3) + (0.5 \times 0.2)$$
$$P(A) = 0.45 + 0.21 + 0.10 = 0.76$$

✅ So, there's a **76% total chance** that a pizza is delivered on time.

```
┌───────────────┐
| John (50%)    |───► 90% on time ⇒ 0.9 × 0.5 = 0.45
└───────────────┘

┌───────────────┐
| Mike (30%)    |───► 70% on time ⇒ 0.7 × 0.3 = 0.21
└───────────────┘

┌───────────────┐
| Anna (20%)    |───► 50% on time ⇒ 0.5 × 0.2 = 0.10
└───────────────┘

➤ Add all contributions:
  0.45 + 0.21 + 0.10 = **0.76**

✅ Final Answer: **There’s a 76% chance a pizza is delivered on time.**
```

In AI / Expert Systems Context: Medical Example

Suppose:

- **Disease A** and **Disease B** both can cause **fever** (Event F)
- P(F | A) = 0.8 (chance of fever if a patient has Disease A)
- P(F | B) = 0.6 (chance of fever if a patient has Disease B)
- P(A) = 0.3 (30% of patients have Disease A)
- P(B) = 0.7 (70% have Disease B)

💡 **Question:**

What is the total probability that a patient has a fever?

$$P(F) = (0.8 \times 0.3) + (0.6 \times 0.7) = 0.24 + 0.42 = 0.66$$

✅ So, **66% of patients** have a fever based on disease distribution.

Bayes' Theorem Used in Machine Learning

Application	Where Bayes is Used	Purpose
Naïve Bayes Classifier	Text classification, spam detection	Fast, interpretable ML classification
Bayesian Networks	Medical, financial expert systems	Probabilistic reasoning
Bayesian Inference	Parameter learning from data	Model updating
Bayesian Optimization	Neural network tuning	Efficient hyperparameter search
Bayesian Deep Learning	Self-driving, medical imaging	Prediction with confidence
VAEs & Bayesian GANs	Generative deep learning	Encoding uncertainty in latent space

Issue with classical system

- Classic rule-based systems (like: IF fever AND cough THEN flu) work well when facts are 100% certain.
- But real-world knowledge is often uncertain or incomplete. So we assign probabilities to facts and rules to model uncertainty.

Term	Meaning
Fact	A piece of information (e.g., "Patient has fever")
Rule	A logical statement (e.g., "IF fever AND cough THEN flu")
Probability in Fact	How likely a fact is true
Probability in Rule	How reliable the rule itself is (i.e., how confident we are in the conclusion **given the facts**)

Probability in Facts

Definition:

This is the probability that a specific condition or fact is true.

Example:

- The system gets a patient record that says:
  - Fever: Yes, with probability 0.8
  - Cough: Yes, with probability 0.9

This means:

"We are 80% sure the patient has a fever and 90% sure they have a cough."

Probability in Rules

Definition:

This is the confidence level in the rule itself, based on expert knowledge or data.

Example Rule:

If a patient has both symptoms, there's an 85% chance they have the flu."

This is also called rule certainty factor or confidence factor.

```
IF fever AND cough THEN flu (with 85% confidence)
```

How to Combine Probabilities in Facts and Rules

Case: Patient has:

- Fever: 80% sure (0.8)
- Cough: 90% sure (0.9)
- Rule: If both, then flu with 85% certainty (0.85)

How to calculate:

To estimate the probability of flu, we multiply:

So, the estimated probability that the patient has the flu is 61.2%.

Example: Security Expert System

Rule: IF unusual login time AND foreign IP THEN suspicious activity (Rule Confidence = 0.9)

📌 Input Facts:

- Unusual login time = 0.7
- Foreign IP address = 0.6

Compute:

Result: There is a 37.8% probability that the activity is suspicious.

$$P(\text{flu}) = P(\text{fever}) \times P(\text{cough}) \times P(\text{rule confidence})$$

$$P(\text{flu}) = 0.8 \times 0.9 \times 0.85 = 0.612$$

$$P(\text{suspicious}) = 0.7 \times 0.6 \times 0.9 = 0.378$$

Why Probability in facts and rules important?

In real-life AI expert systems, we:

- Rarely have perfect data
- Must handle uncertainty
- Need to make decisions based on probabilities and degrees of truth

Cumulative Probability

Definition:

- Cumulative Probability is the probability that a random variable is less than or equal to a certain value.
- In short, it adds up probabilities from the start of the distribution up to a certain point.

In Simple Terms:

- "It's the total chance that an event happens at or before a specific point."
- Think of it as "running total of probabilities."

Mathematically:

If X is a random variable and x is a value, then: $P(X \leq x)$

This is the cumulative probability up to value x.

Example 1: Rolling a Die

- Sample space: {1, 2, 3, 4, 5, 6}
- Each number has a probability of $\frac{1}{6} \approx 0.167$

🔍 Question: What is the **cumulative probability** of rolling a number ≤ 4?

$$P(X \leq 4) = P(1) + P(2) + P(3) + P(4)$$

$$= \frac{1}{6} + \frac{1}{6} + \frac{1}{6} + \frac{1}{6} = \frac{4}{6} = 0.667$$

✅ So, there's a **66.7% chance** of rolling a 4 or less.

📈 **Where is this used in Computer Science?**

Cumulative probability is **everywhere** in CS, especially in:

Area	How It's Used
AI / Expert Systems	Combining multiple uncertain rules and evaluating total likelihood
Data Mining	Cumulative distribution to assess user behavior or risk
Cybersecurity	Anomaly detection based on cumulative event probabilities
Machine Learning	Probability thresholds in classification models (e.g., logistic regression)
Operating Systems	Job scheduling – predicting cumulative wait times

Example 2: Spam Classification

Problem:

You're building a spam filter that checks **word frequency** in emails.

Each suspicious word adds **some probability** to the email being spam.

| Word | P(Spam | Word) | |-------------|----------------| | "win" | 0.4 | | "money" | 0.3 | | "free" | 0.2 |

Goal: Calculate the **cumulative probability** that an email is spam **given all three words** appear.

$$P(\text{Spam} \mid \text{All 3}) = P(\text{win}) + P(\text{money}) + P(\text{free}) = 0.4 + 0.3 + 0.2 = 0.9$$

✅ There's a **90% cumulative chance** that the email is spam if all those words appear.

📌 This is a **simplified form**. Real Naïve Bayes uses **Bayes' Theorem**, but this **illustrates cumulative addition** clearly.

Example 3: Cumulative Distribution Function (CDF)

In statistics, we use **CDF** to represent cumulative probability **graphically**.

📘 **Example: Logistic Regression Model**

In machine learning, a logistic model outputs probabilities like:

Probability	Classification
< 0.5	Not Spam
≥ 0.5	Spam

To **adjust thresholds**, you use **cumulative distributions**:

- If too many messages are **close to 0.5**, you might shift the threshold.
- You analyze **P(Spam ≤ x)** to understand **how confident your model is overall.**

Concept	Meaning
Cumulative Probability	The probability that a value is **less than or equal to** a given value
Used in	AI, ML, expert systems, scheduling, classification
Why it matters	Helps make decisions when **evidence is partial or builds up** over time

What is a Rule-Based System?

Definition:

A rule-based system is an AI system that uses a set of IF-THEN rules to make decisions or draw conclusions.

Structure:

1. Knowledge Base – Stores rules and facts.
2. Inference Engine – Applies logical reasoning to infer new facts.
3. User Interface – For interacting with the system.

📌 Features of Rule-Based Systems:

Feature	Description
Deterministic	Same input gives the same output
Transparent	Easy to explain decisions (can trace rule path)
Fast	Efficient for well-defined problems

❌ Limitation:

They **struggle with uncertainty** – for example:

- What if the symptom is only mild?
- What if data is missing?
- What if multiple diseases match?

Combining Rule-Based System + Bayesian Method

This gives us a hybrid system:

- Use rules to define knowledge structure and logic
- Use Bayesian probability to handle uncertainty

Real-World Hybrid Example (Medical Expert System):

Step 1: Rule-Based

Step 2: Bayesian Enhancement

Add probabilities:

- P(fever | flu) = 0.9
- P(cough | flu) = 0.8
- P(flu) = 0.1
- Combine these using Bayes' theorem to calculate final diagnosis probability

This approach:

- Supports partial truths (symptoms may not be 100% present)
- Can rank diseases by likelihood

- Handles multiple conflicting rules smoothly

```
IF fever AND cough THEN potential_flu
```

Rule Based Bayesian Method

Feature	Rule-Based System	Bayesian Method
Handles uncertainty	❌ No	✅ Yes
Reasoning type	Logical (IF-THEN)	Probabilistic
Output	Deterministic (fixed result)	Probabilistic (confidence score)
Explainability	High (trace rules)	Medium (needs math explanation)

Fuzzy - Introduction

In everyday language, we often use words that aren't exact, like "hot," "cold," or "tall." For example, instead of saying someone is "tall" or "not tall," we might say they're "somewhat tall" or "very tall." Fuzzy logic helps us represent this kind of thinking mathematically.

In fuzzy logic, we use numbers between 0 and 1 to express how true something is:

1 means fully true (100%).

0 means completely false (0%).

Any number between 0 and 1 means partially true (like 0.5 meaning 50% true).

Example:

- Words like young, tall, good or high are fuzzy.
- There is no single quantitative value which defines the term young.
- For some people, age 25 is young, and for others, age 35 is young.
- The concept young has no clean boundary.
- Age 35 has some possibility of being young and usually depends on the context in which it is being considered.

Fuzzy example

📊 Example: Temperature

Let's say we want to describe the concept of "Warm temperature".

In **classical logic**, we might say:

- If temp is **25°C**, it's warm → ✅ (1)
- If temp is **10°C**, it's not warm → ❌ (0)

But in **fuzzy logic**:

- 25°C might be **warm = 1.0**
- 20°C might be **warm = 0.7**
- 15°C might be **warm = 0.3**
- 10°C might be **warm = 0.0**

📐 Membership Function

A **membership function (μ)** maps each input value to a number between **0 and 1**, showing **how much** it belongs to the fuzzy set.

Common shapes:

Function Type	Used for...
Triangular	Simple categories (Cold, Warm, Hot)
Trapezoidal	Smooth ranges (e.g., Medium speed)
Gaussian	Natural distributions

So **fuzzy sets allow gradual membership**, like shades of gray instead of black and white.

Fuzzy Logic (1)

In classical set theory:

- A set A is a subset of a universe X: $A \subseteq X$
- For any element $x \in X$, either:

$$x \in A \quad \text{or} \quad x \notin A$$

A **fuzzy set A** in universe **X** is defined as:

$$A = \{(x, \mu_A(x)) \mid x \in X\}$$

- **x**: an element in the universe
- **μ_A(x)**: degree of membership of x in A (between 0 and 1)

Fuzzy Logic (2)

Let **X = {10, 20, 30, 40}** be temperatures in °C.
Define fuzzy set **A = "Warm"** as:

x (°C)	μ_Warm(x)
10	0.0
20	0.4
30	1.0
40	0.3

So, the fuzzy set is:

$$\text{Warm} = \{(10, 0.0), (20, 0.4), (30, 1.0), (40, 0.3)\}$$

Why Use Multivalued Logic?

Because real-world situations are rarely black or white.

For example:

In classical logic:

- "It is hot" → either True (1) or False (0)

But in reality:

- At 28°C, it may feel partially hot — say 0.7
- At 22°C, it may feel a bit hot — say 0.3

That’s where multivalued logic comes in — to describe this degree of truth.

Crisp vs Fuzzy

Definition	Crisp, well-defined boundaries	Gradual boundaries, allows partial membership
Membership Values	Only `0` (not in set) or `1` (in set)	Any value in `[0, 1]` (partial inclusion)
Membership Function	Not typically used	Defined: $\mu_A(x) : X \to [0, 1]$
Element Test	$x \in A$ or $x \notin A$	$\mu_A(x) = 0.0$ to $\mu_A(x) = 1.0$
Use Case	Binary logic, math, databases	Real-world reasoning, AI, control systems
Example: “Tall”	Height ≥ 180 → Tall (1), else (0)	170 cm = 0.5 tall, 180 cm = 0.8 tall, etc.
Transition Zones	Sharp boundary between sets	Smooth transition between sets
Set Notation	$A = \{x \mid x > 5\}$	$A = \{(x, \mu_A(x)) \mid x \in X\}$
Logic Basis	Boolean logic (True/False)	Fuzzy logic (degrees of truth)

Basic Fuzzy Set Operations (1)

Fuzzy set operations are similar to classical set operations but are based on degrees of membership (values between 0 and 1).

Let:

- $\mu A(x)$: membership of element x in fuzzy set A
- $\mu B(x)$: membership of element x in fuzzy set B

1. Union of Two Fuzzy Sets (A ∪ B)

Definition:

The degree of membership of an element in the union of A and B is the maximum of its membership in A and B.

mu_A∪B(x) = max(mu_A(x), mu_B(x))

x	mu_A(x)	mu_B(x)	mu_A∪B(x)
1.	0.2	0.4	0.4
2.	0.6	0.5	0.6
3.	0.9	0.7	0.9

Basic Fuzzy Set Operations (2)

2. Intersection of Two Fuzzy Sets (A ∩ B)

- The degree of membership of an element in the intersection of A and B is the minimum of its membership in A and B.

mu_A∩B(x) = min(mu_A(x), mu_B(x))

x	mu_A(x)	mu_B(x)	mu_A∩B(x)
1.	0.2	0.4	0.2
2.	0.6	0.5	0.5
3.	0.9	0.7	0.7

3. Complement of a Fuzzy Set (¬A)

- The degree of membership of an element in the complement of A is 1 - mu_A(x).

mu_¬A(x) = 1 - mu_A(x)

x	mu_A(x)	mu_¬A(x)
1.	0.2	0.8
2.	0.6	0.4
3.	0.9	0.1

```
Operation       | Formula
----------------|----------------------------------
Union (A ∪ B)    | mu_A∪B(x) = max(mu_A(x), mu_B(x))
Intersection     | mu_A∩B(x) = min(mu_A(x), mu_B(x))
Complement (¬A) | mu_¬A(x)  = 1 - mu_A(x)
```

Membership functions

- If we plot x on the horizontal axis (temperature), and μ_A(x) on the vertical axis, we get a membership curve — often triangular or trapezoidal.

Common forms:

1. **Triangular**:

$$\mu_A(x) = \begin{cases} 0, & x \le a \text{ or } x \ge c \\ \frac{x-a}{b-a}, & a < x \le b \\ \frac{c-x}{c-b}, & b < x < c \end{cases}$$

2. **Trapezoidal**:

$$\mu_A(x) = \begin{cases} 0, & x \le a \text{ or } x \ge d \\ \frac{x-a}{b-a}, & a < x \le b \\ 1, & b < x \le c \\ \frac{d-x}{d-c}, & c < x < d \end{cases}$$

3. **Gaussian**:

$$\mu_A(x) = e^{-\frac{(x-c)^2}{2\sigma^2}}$$

Like classical sets, fuzzy sets support operations — **but with degree logic**:

1. **Union**:

$$\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x))$$

2. **Intersection**:

$$\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x))$$

3. **Complement**:

$$\mu_{\neg A}(x) = 1 - \mu_A(x)$$

Fuzzy Logic – Simple Example (1)

📃 Step 1: Define the Universe

Let's say:

- Temperature ranges from 0°C to 40°C
- We define 3 fuzzy sets:
  - Cold
  - Warm
  - Hot

📐 Step 2: Membership Functions (Triangular)

We define fuzzy sets like this:

- Cold : 0 to 20°C
- Warm : 15 to 30°C
- Hot : 25 to 40°C

Let’s assume:

- At 15°C:
  - μ_Cold = 1.0
  - μ_Warm = 0.0
  - μ_Hot = 0.0
- At 22°C:
  - μ_Cold = 0.2
  - μ_Warm = 0.8
  - μ_Hot = 0.0
- At 30°C:
  - μ_Cold = 0.0
  - μ_Warm = 0.5
  - μ_Hot = 0.5
- At 38°C:
  - μ_Cold = 0.0
  - μ_Warm = 0.0
  - μ_Hot = 1.0

Fuzzy Logic – Simple Example (2)

Step 3: Define Rules (Fuzzy Inference)

Let's use simple rules:

- IF temperature is cold → THEN heater power = high

- IF temperature is warm → THEN heater power = medium
- IF temperature is hot → THEN heater power = low

Step 4: Apply Fuzzy Logic (Numerical Example)

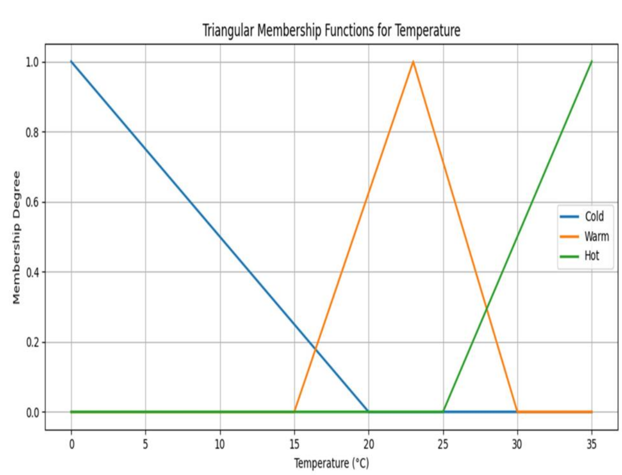
Let’s say the temperature is 22°C.

From Step 2:

- μ_Cold = 0.2
- μ_Warm = 0.8
- μ_Hot = 0.0

Apply rules:

- Rule 1: 0.2 → High heating
- Rule 2: 0.8 → Medium heating
- Rule 3: 0.0 → Low heating



Fuzzy Logic – Simple Example (3)

Step 5: Defuzzification (Crisp Output)

Assume:

- High heating = 80%
- Medium = 50%
- Low = 20%

We calculate a weighted average:

Output heating = (0.2×80 + 0.8×50 + 0.0×20) / (0.2 + 0.8 + 0.0)

= (16 + 40 + 0) / 1.0

= 56% ➡️ Final Output: Apply **56% heating power**

What is a Linguistic Variable?

A linguistic variable is a variable whose values are words or sentences (in natural language) rather than numbers.

Instead of temperature = 25°C

We use: temperature = "cold", "warm", "hot“

Each of these words (“cold”, “warm”, “hot”) is a linguistic value, and each has a fuzzy set associated with it.

Let's say we want to control a fan based on room temperature.

- **Linguistic Variable**: `temperature`
- **Linguistic Values**: `cold`, `warm`, `hot`

Each of these has a **membership function**:

Temperature (°C)	μ_cold	μ_warm	μ_hot
10	1.0	0.0	0.0
20	0.4	0.6	0.0
25	0.0	1.0	0.2
35	0.0	0.1	0.9

What is a Hedge?

Hedges are modifiers applied to fuzzy sets (linguistic values) to make them stronger or weaker — similar to adverbs in language.

Examples:

- “Very hot”
- “Somewhat warm”
- “Extremely cold”
- “Not warm”

Common Fuzzy Hedges (Mathematical)

Example

Suppose: $\mu\_hot(temperature = 30°C) = 0.6$

Then:

- $\mu\_very\_hot = (0.6)^2 = 0.36$ → less intense
- $\mu\_somewhat\_hot = \sqrt{0.6} \approx 0.77$ → more intense
- $\mu\_not\_hot = 1 - 0.6 = 0.4$ → opposite degree

Hedge	Effect	Formula (if μ is the original degree)
Very	Makes the set **sharper**	μ^2
Somewhat	Makes the set **flatter**	$\sqrt{\mu}$
More or less	Similar to "somewhat"	μ^0.5
Not	Logical negation	1 - μ

So, "very hot" becomes 0.36, meaning:

It's somewhat hot, but not yet very hot.

- Something that is hot to a degree of 0.6 is not fully hot.
- So it cannot be "very hot" — hence a lower membership.

Example of Hedge

Example with μ = 0.6:

Expression	Result
μ_hot	0.6
μ_very_hot	0.36
μ_somewhat_hot	0.77
μ_not_hot	0.4

What Is a Fuzzy Proposition?

- A fuzzy proposition is a statement that expresses a condition using linguistic terms and fuzzy logic.
- It's similar to a normal logic statement (like "The temperature is hot") — but in fuzzy logic, the truth of the statement is a value between 0 and 1, not just true or false.

In Fuzzy Logic:

Fuzzy Proposition:

“The temperature is hot.”

- Temp = 28°C → μ_hot(28) = 0.7
- Temp = 22°C → μ_hot(22) = 0.2

So the truth value of the proposition depends on the membership degree.

Structure of a Fuzzy Proposition

<Variable> is <Linguistic Term>

Example:

- "Temperature is warm"
- "Speed is very slow"
- "Soil moisture is dry“
- Each linguistic term (like "warm") is a fuzzy set, and the proposition's truth value depends on how much the current value (e.g., temperature) belongs to that fuzzy set.

Compound Fuzzy Propositions

We can combine fuzzy propositions using logical operators, just like in classical logic, but with fuzzy equivalents

Example:

“Temperature is warm AND humidity is high”

If:

- μ_warm(temperature = 28°C) = 0.8
- μ_high_humidity(humidity = 75%) = 0.6

Then:

- μ_proposition = min(0.8, 0.6) = 0.6, So the compound proposition is 0.6 true.

Scenario	Fuzzy Proposition	Used For
Air conditioning	“Temperature is hot”	Adjust cooling power
Washing machine	“Clothes are very dirty”	Set washing cycle time
Autonomous vehicle	“Speed is slightly high”	Reduce acceleration
Irrigation system	“Soil is dry AND temperature is high”	Turn on water supply
Customer service bot	“User is not satisfied”	Escalate to human support

Fuzzy logic system Speed of fan based on Temperature (1)

Step 1: Define Input and Output Variables

- Input Variable: Temperature (°C) Fuzzy sets:
  - Cold
  - Warm
  - Hot
- Output Variable: Fan Speed (%) Fuzzy sets:
  - Low
  - Medium
  - High

Step 2: Define Membership Functions (Triangular) - Fuzzification

🔷 Temperature (range: 0°C to 40°C)

- Cold: triangle with peak at 0°C, ends at 20°C → [0, 0, 20]
- Warm: triangle with peak at 25°C, spans 15°C–35°C → [15, 25, 35]
- Hot: triangle with peak at 40°C, starts at 30°C → [30, 40, 40]

🔷 Fan Speed (range: 0% to 100%)

- Low: triangle with peak at 0% → [0, 0, 50]
- Medium: triangle centered at 50% → [30, 50, 70]
- High: triangle with peak at 100% → [60, 100, 100]

Fuzzy logic system Speed of fan based on Temperature (2)

Step 3: Define Fuzzy Inference Rules

1. IF temperature is cold → THEN fan speed is low
2. IF temperature is warm → THEN fan speed is medium
3. IF temperature is hot → THEN fan speed is high

These are simple, intuitive, and human-like rules.

Step 4: Fuzzification

- Convert the input temperature value into membership degrees.
- Example: If temperature = 28°C:
    - μ_cold = 0.0
    - μ_warm = 0.7
    - μ_hot = 0.3

Step 5: Inference

- Rule 2 (warm): Fires at 0.7 → contributes 0.7 to medium speed
- Rule 3 (hot): Fires at 0.3 → contributes 0.3 to high speed

Fuzzy logic system Speed of fan based on Temperature (3)

Step 6: Aggregation

- Combine the fuzzy outputs from all rules into a single fuzzy set.

Step 7: Defuzzification

- Convert the fuzzy output set into a single crisp value (e.g., 67%) using centroid or other defuzzification methods.

Step 8: Output Result

Return the crisp fan speed based on

the input temperature.

Example:

- Input temperature: 28°C
- Output fan speed: 67.5%

Step	Description
Define Input	Temperature (°C)
Define Output	Fan Speed (%)
Fuzzy Sets	Cold, Warm, Hot → Low, Medium, High
Rules	Simple IF-THEN rules
Fuzzify	Map temperature to fuzzy values
Apply Inference	Evaluate rule activations
Aggregate & Defuzzify	Combine and convert to a crisp value
Output	Final fan speed recommendation

Fuzzy rule set for a chatbot that gives polite, friendly responses (1)

🎯 Goal:

Make the chatbot respond more politely and warmly when the user seems upset or neutral, and more casually when the user is happy.

🔷 Input (Fuzzy Variable): User Mood

Fuzzy sets:

- Angry
- Neutral
- Happy

🔷 Output (Fuzzy Variable): Chatbot Response Tone

Fuzzy sets:

- Very Polite
- Polite
- Casual

Fuzzy rule set for a chatbot that gives polite, friendly responses (2)

🧠 Fuzzy Rule Set:

Rule No.	Fuzzy Rule
1	**IF** user mood is angry **THEN** response tone is very polite
2	**IF** user mood is neutral **THEN** response tone is polite
3	**IF** user mood is happy **THEN** response tone is casual

User Message	Detected Mood	Chatbot Tone	Sample Response
"This app is so frustrating!"	Angry	Very Polite	"I'm really sorry you're facing trouble. Let me help!"
"Okay, what's next?"	Neutral	Polite	"Sure! Here's what we can do next 😊 "
"Hey, that worked perfectly!"	Happy	Casual	"Awesome! Glad it worked! 🎉 "