



UNIT-III: Handling Uncertainty & Natural Language Processing

Complete Study Notes with Examples, Code, Diagrams & Resources

TABLE OF CONTENTS

1. Handling Uncertainty
 2. Natural Language Processing (NLP)
 3. Code Implementations
 4. Images & Infographics
 5. Reference Sources
-

1. HANDLING UNCERTAINTY

Why Uncertainty?

Uncertainty: Situation where information is incomplete, unreliable, or ambiguous.^[1]

Sources of Uncertainty:^[1]

Source	Description	Example
Incomplete Data	Missing information	Patient symptom not reported
Unreliable Data	Sensor errors, noise	Temperature sensor reads 5°C too high
Ambiguity	Multiple interpretations	"bank" = river vs money
Vagueness	unclear boundaries	"hot" temperature = ?
Randomness	Probabilistic events	Coin toss, dice roll

Why AI Needs Uncertainty Handling:

- ❌ Real world is uncertain (not deterministic)
- ❌ Sensors have noise/errors
- ❌ Human knowledge is vague ("probably", "maybe")
- ✅ AI must reason with incomplete info

Non-Monotonic Reasoning

Definition: Reasoning where adding new facts can **retract** previous conclusions.^[2]

Monotonic vs Non-Monotonic:

Property	Monotonic Reasoning	Non-Monotonic Reasoning
New Facts	Never retract conclusions	Can retract conclusions
Truth	Always preserved	Can change
Example	Math proofs	Common sense reasoning
Use Case	Formal logic	Real-world AI

Monotonic Example (Math):

Fact 1: $2 + 2 = 4$

Fact 2: All even numbers are divisible by 2

↓ Add new fact

Fact 3: 4 is even

Conclusion: 4 is divisible by 2 ✓ (never retracted)

Non-Monotonic Example (Medical):

Fact 1: Patient has fever

→ Conclusion: Patient has flu ✓

Fact 2: Patient has malaria (NEW!)

→ Conclusion: Patient does NOT have flu ✗ (retracted!)

Non-Monotonic Logic Features:^[1]

- Makes assumptions **unless contradicted**
- Uses **default logic**: "Birds fly" (except penguins)
- Supports **belief revision**

Example: Default Logic:^[1]

Default Rule: $\text{Bird}(x) \rightarrow \text{Fly}(x)$

Exception: $\text{Penguin}(x) \rightarrow \neg\text{Fly}(x)$

Fact 1: $\text{Bird}(\text{Tweety})$

→ Conclusion: $\text{Fly}(\text{Tweety})$ ✓

Fact 2: $\text{Penguin}(\text{Tweety})$

→ Conclusion: $\neg\text{Fly}(\text{Tweety})$ ✗ (retracted!)

Probabilistic Reasoning

Definition: Using probability theory to represent and reason with uncertainty.^[1]

Key Concepts:

Concept	Symbol	Meaning	Value Range	
Probability	P(A)	Likelihood of A	0 to 1	
Conditional	P(A B)	P(A AND B)	P(A) given B true	0 to 1
Joint	P(A, B)	P(A AND B)	0 to 1	
Marginal	P(A)	P(A) alone	0 to 1	

Probability Rules:^[1]

1. **Non-negativity:** $P(A) \geq 0$
2. **Normalization:** $P(True) = 1, P(False) = 0$
3. **Addition:** $P(A \vee B) = P(A) + P(B) - P(A \wedge B)$
4. **Multiplication:** $P(A \wedge B) = P(A|B) \cdot P(B)$
5. **Complement:** $P(\neg A) = 1 - P(A)$

Example:

$$P(\text{rain}) = 0.3$$

$$P(\text{sunny}) = 0.7$$

$$P(\text{rain} \vee \text{sunny}) = 0.3 + 0.7 = 1.0 \quad \checkmark$$

$$P(\neg \text{rain}) = 1 - 0.3 = 0.7 \quad \checkmark$$

Bayes' Theorem

Most important for probabilistic reasoning.^{[3][4]}

Formula:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \quad [5]$$

Where:

- $P(A|B)$: Probability of A **given** B (posterior)
- $P(B|A)$: Probability of B **given** A (likelihood)
- $P(A)$: Prior probability of A
- $P(B)$: Prior probability of B

Bayes' Theorem Example 1: Spam Detection^{[4][3]}

Problem: Is email spam if it contains "offer"?

Given:

$P(\text{spam}) = 0.3$ (30% of emails are spam)

$P(\text{"offer"}|\text{spam}) = 0.8$ (80% of spam has "offer")

$P(\text{"offer"}|\text{not spam}) = 0.1$ (10% of normal has "offer")

Find: $P(\text{spam}|\text{"offer"})$

Solution:

$$\begin{aligned}P(\text{"offer"}) &= P(\text{"offer"}|\text{spam}) \cdot P(\text{spam}) + P(\text{"offer"}|\text{not spam}) \cdot P(\text{not spam}) \\ &= 0.8 \cdot 0.3 + 0.1 \cdot 0.7 \\ &= 0.24 + 0.07 = 0.31\end{aligned}$$

$$\begin{aligned}P(\text{spam}|\text{"offer"}) &= \frac{P(\text{"offer"}|\text{spam}) \cdot P(\text{spam})}{P(\text{"offer"})} \\ &= \frac{0.8 \cdot 0.3}{0.31} \\ &= \frac{0.24}{0.31} \\ &= 0.774 \checkmark \text{ (77.4\% chance it's spam)}\end{aligned}$$

Bayes' Theorem Example 2: Medical Diagnosis^[3]

Problem: Patient tests positive for disease. What's probability of disease?

Given:

$P(\text{disease}) = 0.01$ (1% population has disease)

$P(\text{positive}|\text{disease}) = 0.99$ (99% sensitivity)

$P(\text{positive}|\text{no disease}) = 0.01$ (1% false positive)

Find: $P(\text{disease}|\text{positive})$

Solution:

$$P(\text{positive}) = 0.99 \cdot 0.01 + 0.01 \cdot 0.99$$

$$= 0.0099 + 0.0099 = 0.0198$$

$$P(\text{disease}|\text{positive}) = 0.99 \cdot 0.01 / 0.0198$$

$$= 0.0099 / 0.0198$$

$$= 0.5 \checkmark \text{ (50\% chance despite positive test!)}$$

Why only 50%? Because disease is rare (1%), so most positive tests are false alarms.

Bayes' Rule for Multiple Events:

$$P(A_i|B) = \frac{P(B|A_i) \cdot P(A_i)}{\sum_j P(B|A_j) \cdot P(A_j)} \quad [3]$$

Certainty Factors and Rule-Based Systems

Certainty Factor (CF): Measure of belief strength in rule-based systems. [1]

Formula:

$$CF = \frac{P(H|E) - P(H)}{1 - P(H)}$$

Where:

- H : Hypothesis
- E : Evidence
- CF range: -1 (disbelief) to +1 (belief)

Certainty Factor Example: [1]

Rule: IF fever AND cough \rightarrow THEN flu (CF = 0.8)

Evidence:

fever = true (confidence = 0.9)

cough = true (confidence = 0.7)

Combined evidence confidence:

$$CF_evidence = 0.9 \times 0.7 = 0.63$$

Final conclusion confidence:

$CF_{flu} = CF_{evidence} \times CF_{rule}$
 $= 0.63 \times 0.8$
 $= 0.504$ ✓ (50.4% belief in flu)

Rule-Based System with Certainty:^[1]

R1: IF symptom_A THEN disease_X (CF = 0.9)
R2: IF symptom_B THEN disease_X (CF = 0.7)
R3: IF disease_X THEN treat_Y (CF = 0.85)

Evidence:

symptom_A = true (0.8)
symptom_B = true (0.6)

Inference:

$CF_{X_from_A} = 0.8 \times 0.9 = 0.72$
 $CF_{X_from_B} = 0.6 \times 0.7 = 0.42$
 $CF_{X_combined} = \max(0.72, 0.42) = 0.72$ ✓

$CF_{treat} = 0.72 \times 0.85 = 0.612$ ✓ (61.2% belief in treatment Y)

Bayesian Networks

Definition: Probabilistic graphical model using directed acyclic graph (DAG).^{[6][1]}

Components:^[1]

1. **Directed Acyclic Graph (DAG):** Nodes = variables, Edges = dependencies
2. **Conditional Probability Table (CPT):** $P(\text{node} \mid \text{parents})$

Bayesian Network Example: Medical Diagnosis^[1]

Graph:

[Disease] → [Test_Result]
↓
[Symptom]

CPTs:

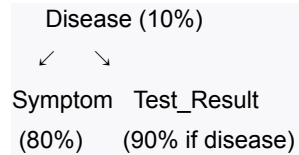
$P(\text{Disease}) = 0.1$ (10% have disease)
 $P(\text{Test_Result}=\text{positive} \mid \text{Disease}=\text{true}) = 0.9$

$P(\text{Test_Result}=\text{positive} \mid \text{Disease}=\text{false}) = 0.2$

$P(\text{Symptom} \mid \text{Disease}=\text{true}) = 0.8$

$P(\text{Symptom} \mid \text{Disease}=\text{false}) = 0.1$

Visual:



Properties:^[1]

Property	Description
Directed	Edges have direction (cause → effect)
Acyclic	No cycles (can't loop back)
Conditional Independence	Node independent of non-descendants given parents

Advantages:^[1]

- Visual representation
- Efficient inference
- Handles missing data
- Combines expert knowledge + data

Applications:^[1]

- Medical diagnosis
- Spam filtering
- Fault detection
- Decision support

Probabilistic Inference:

Query: $P(\text{Disease} \mid \text{Test}=\text{positive}, \text{Symptom}=\text{true})$

Use Bayes' rule + conditional independence:

$$P(D|T,S) = P(T|D) \cdot P(S|D) \cdot P(D) / P(T,S)$$

Dempster-Shafer Theory

Definition: Mathematical framework for combining evidence with incomplete information.^{[7][8]}

Key Difference from Probability:

Aspect	Probability	Dempster-Shafer
Requirement	Prior probabilities needed	No priors needed
Uncertainty	Single value	Interval [belief, plausibility]
Ignores	Must assign to all outcomes	Can assign to "unknown"

Key Concepts.^{[8][7]}

Term	Symbol	Definition
Mass Function	$m(A)$	Degree of belief assigned to set A
Belief	$Bel(A)$	Sum of masses for all subsets of A
Plausibility	$Pl(A)$	Sum of masses for all sets overlapping A
Belief Interval	$[Bel(A), Pl(A)]$	Range of possible belief

Formulas:^[7]

$$Bel(A) = \sum_{B \subseteq A} m(B) \quad [9]$$

$$Pl(A) = \sum_{B \cap A \neq \emptyset} m(B) = 1 - Bel(\neg A) \quad [10]$$

Dempster-Shafer Example: Murder Case^[7]

Hypotheses: {Alice, Bob, Carol}

Evidence 1: 指纹 at scene → Alice (m = 0.6)

Evidence 2: Witness saw tall person → Bob (m = 0.3)

Unknown: m({Alice, Bob, Carol}) = 0.1

Calculate for Alice:

$$Bel(Alice) = m(\{Alice\}) = 0.6$$

$$Pl(Alice) = m(\{Alice\}) + m(\{Alice, Bob, Carol\}) \\ = 0.6 + 0.1 = 0.7$$

Belief Interval: [0.6, 0.7] ✓

Calculate for Bob:

$$Bel(Bob) = m(\{Bob\}) = 0.3$$

$$Pl(Bob) = 0.3 + 0.1 = 0.4$$

Belief Interval: [0.3, 0.4] ✓

Combining Evidence (Dempster's Rule):^[8]

$$m_{12}(A) = \frac{\sum_{B \cap C = A} m_1(B) \cdot m_2(C)}{1 - K}$$

Where $K = \sum_{B \cap C = \emptyset} m_1(B) \cdot m_2(C)$ (conflict)

Applications:^[8]

- Sensor fusion (combine multiple sensors)
- Expert systems (combine expert opinions)
- Risk assessment
- Decision making under uncertainty

Introduction to Fuzzy Logic

Definition: Logic handling **degrees of truth** (partial membership).^{[1][11]}

Fuzzy vs Boolean Logic:^[11]

Concept	Boolean Logic	Fuzzy Logic
Values	0 or 1 only	Any value 0 to 1
Truth	Absolute	Partial
Use Case	Exact systems	Real-world uncertain
Example	Hot (1) or Not Hot (0)	Slightly Hot (0.3), Warm (0.7)

Fuzzy Logic History:^[1]

- First used: 1965 by Lotfi Zadeh (UC Berkeley)
- Purpose: Handle distinctions neither true nor false
- Key idea: Represent **degree of belongingness** to set

Why Fuzzy Logic?:^{[1][11]}

- Human reasoning is approximate ("probably", "maybe")
- Real-world concepts have unclear boundaries ("hot", "tall")
- Better for control systems (AC, washing machines)

Fuzzy Set Definition & Types

Fuzzy Set Definition:^[12]

A fuzzy set A on universe X is a collection of ordered pairs:

$$A = \{(x, \mu_A(x)) | x \in X\}^{\text{[13]}}$$

Where:

- x : Element
- $\mu_A(x)$: Membership value (0 to 1)
- $\mu_A(x) = 0$: No membership
- $\mu_A(x) = 1$: Full membership

Fuzzy Set Example:^[12]

Universe: Temperature = {0, 10, 20, 30, 40, 50}°C

Fuzzy Set "Warm":

A = {

(0°C, 0.0),

(10°C, 0.2),

(20°C, 0.6),

(30°C, 1.0),

(40°C, 0.7),

(50°C, 0.3)

}

Interpretation:

30°C is 100% warm

20°C is 60% warm

50°C is 30% warm

Crisp vs Fuzzy Set:^[12]

Crisp Set "Hot" (threshold = 40°C):

40°C → 1 (hot)

39°C → 0 (not hot) ✗ abrupt change

Fuzzy Set "Hot":

40°C → 1.0 (fully hot)

35°C → 0.7 (mostly hot)

30°C → 0.3 (slightly hot) ✗ smooth transition ✓

Types of Fuzzy Sets:^[11]

Type	Definition	Example	
Normal	$\max(\mu(x)) = 1$	Standard fuzzy set	
Convex	$\mu(\text{midpoint}) \geq \min(\mu(a), \mu(b))$	Bell-shaped curve	
Symmetric	$\mu(x) = \mu(-x)$ around center	Gaussian	
Alpha-cut	$\{x$	$\mu(x) \geq \alpha\}$	Elements with membership $\geq \alpha$
Empty	$\mu(x) = 0$ for all x	No membership	
Whole	$\mu(x) = 1$ for all x	Full membership	

Membership Functions

Definition: Function mapping input to membership degree (0 to 1).^{[12][11]}

Common Types of Membership Functions:^{[14][11]}

Type	Formula	Shape	Use Case
Singleton	$\mu(x) = 1$ if $x = c$, else 0	Single point	Crisp values
Triangular	$\mu(x) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a < x < b \\ 0 & x \geq b \end{cases}$	Triangle	Simple control
Trapezoidal	Similar to triangular but flat top	Trapezoid	Ranges
Gaussian	$\mu(x) = e^{-\frac{(x-c)^2}{2\sigma^2}}$	Bell curve	Smooth transitions
S-function	Polynomial sigmoid	S-shaped	Increasing membership

Z-functio n	1 - S-function	Z-shaped	Decreasing membership
------------------------	----------------	----------	--------------------------

Triangular Membership Example:^[11]

Set: Warm = triangular(20, 30, 40)

Calculate $\mu_{\text{warm}}(32^\circ\text{C})$:

$$\mu = (40 - 32) / (40 - 30) = 8/10 = 0.8 \quad \checkmark$$

Interpretation: 32°C is 80% warm

Visual Comparison:

Triangular: Trapezoidal: Gaussian:



MATLAB Commands:^[14]

```
% Triangular
trimf(x, [20, 30, 40])

% Gaussian
gaussmf(x, [30, 5])

% Trapezoidal
trapmf(x, [20, 30, 40, 50])
```

Designing a Fuzzy Set for an Application

Step-by-Step Process:^[11]

Example: Temperature-based Fan Control System^[11]

Step 1: Define Universe of Discourse

Input: Temperature = [0°C, 50°C]

Output: Fan Speed = [0%, 100%]

Step 2: Define Fuzzy Sets

Temperature Sets:

- Cold = triangular(0, 10, 20)
- Warm = triangular(20, 30, 40)
- Hot = triangular(30, 40, 50)

Fan Speed Sets:

- Low = triangular(0, 20, 40)
- Medium = triangular(40, 60, 80)
- High = triangular(80, 90, 100)

Step 3: Define Membership Functions^[11]

```
def triangular_membership(x, a, b, c):
```

```
    if x <= a or x >= c:
```

```
        return 0.0
```

```
    elif a < x <= b:
```

```
        return (x - a) / (b - a)
```

```
    elif b < x < c:
```

```
        return (c - x) / (c - b)
```

```
# Example: Temperature = 32°C
```

```
mu_warm = triangular_membership(32, 20, 30, 40) # = 0.8
```

```
mu_hot = triangular_membership(32, 30, 40, 50) # = 0.2
```

Step 4: Create Rule Base^[11]

Rule 1: IF temperature is cold → fan = low

Rule 2: IF temperature is warm → fan = medium

Rule 3: IF temperature is hot → fan = high

Step 5: Fuzzification^[11]

Input: Temperature = 32°C

Fuzzify:

$$\mu_{\text{cold}}(32) = 0.0$$

$$\mu_{\text{warm}}(32) = (40-32)/(40-30) = 0.8$$

$$\mu_{\text{hot}}(32) = (32-30)/(40-30) = 0.2$$

Result: 80% Warm, 20% Hot

Step 6: Inference^[11]

Apply Rules:

Rule 2: warm \rightarrow medium (confidence = 0.8)

Rule 3: hot \rightarrow high (confidence = 0.2)

Fuzzy Output:

$$\mu_{\text{medium}} = 0.8$$

$$\mu_{\text{high}} = 0.2$$

Step 7: Defuzzification^[11]

Convert to Crisp Output:

Assume: Medium = 50%, High = 80%

Centroid Method:

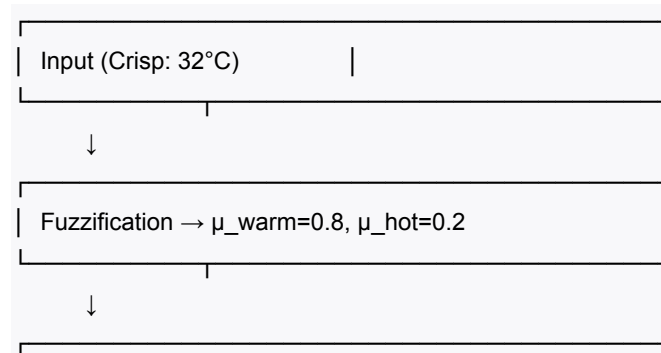
$$\text{Output} = (0.8 \times 50 + 0.2 \times 80) / (0.8 + 0.2)$$

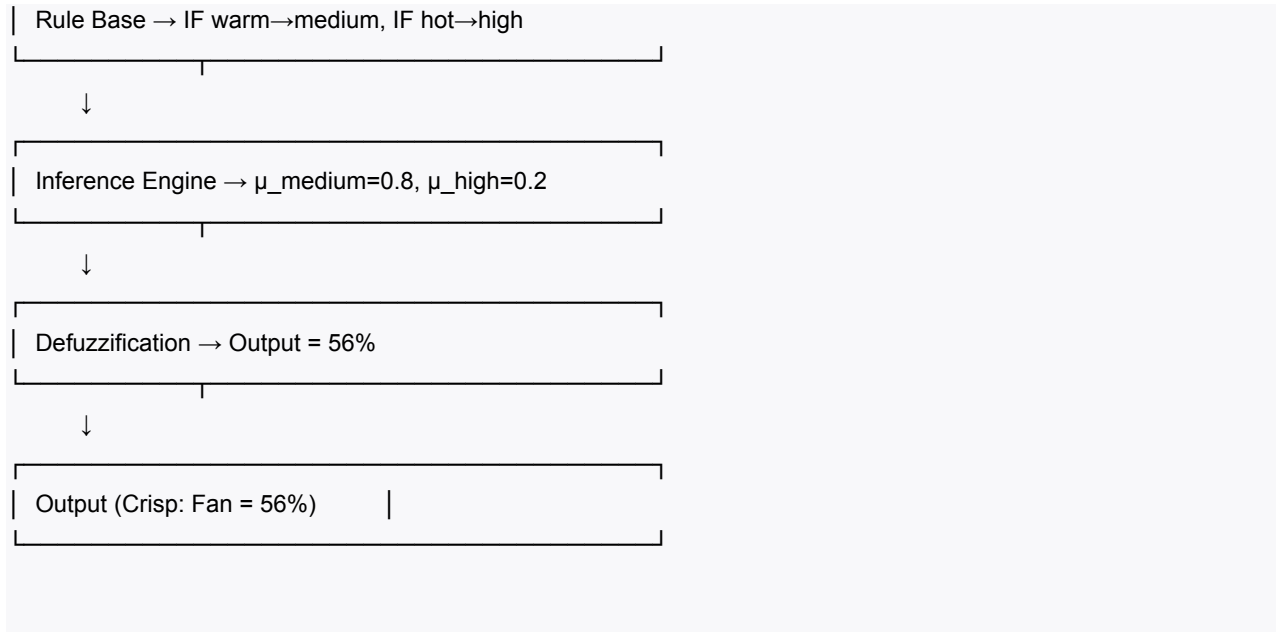
$$= (40 + 16) / 1.0$$

$$= 56\% \checkmark$$

Final: Fan speed = 56%

Fuzzy Logic System Architecture:^[11]





Applications of Fuzzy Logic:^[11]

Domain	Application
Aerospace	Spacecraft altitude control
Automotive	Speed control, traffic management
Industrial	pH control, drying, distillation
AI	NLP, decision support
Consumer	AC, washing machines, cameras

Advantages:^[11]

- Handles noisy/imprecise data
- Easy to design
- Mimics human reasoning
- Low memory/computational cost

Disadvantages:^[11]

- ✗ May produce ambiguous solutions
 - ✗ Difficult to verify correctness
 - ✗ Less accurate for precise data
-

2. NATURAL LANGUAGE PROCESSING (NLP)

Introduction to NLP

Definition: Field enabling computers to understand, analyze, and interact with human language.^[15]

What NLP Does:

- Helps computers understand human language
- Processes structure AND meaning
- Uses multiple analysis phases
- Applied in: chatbots, translation, sentiment analysis, search

NLP Applications:

Application	Example
Chatbots	Customer support automation
Translation	Google Translate
Sentiment Analysis	"This product is amazing!" -> Positive
Search	Understanding query intent
Text Summarization	Auto-generate article summaries

Lexical and Morphological Analysis

Initial phases identifying words and structure.^[15]

Lexical Analysis:^[15]

Tokenization: Split text into tokens (words)

Sentence: "I am reading a book."

Tokens: ["I", "am", "reading", "a", "book"]

Part-of-Speech (POS) Tagging: Assign grammatical roles

"I" → Pronoun (PRP)

"am" → Verb (VBP)

"reading" → Verb (VBG)

"a" → Article (DT)

"book" → Noun (NN)

Morphological Analysis:^[15]

Stemming: Reduce to root form

"running" → "run"

"flies" → "fli"

"better" → "bet"

Lemmatization: Convert to base form (considering context)

"running" → "run"

"flies" → "fly"

"better" → "good" ✓

Why Important:

- Improves parsing accuracy
- Supports translation
- Enables word normalization

Syntactic Processing (Parsing)

Definition: Analyzing grammatical structure according to grammar rules.^{[16][15]}

What It Does:

- Identifies subject, verb, object
- Creates parse tree
- Shows relationships between words

Syntactic Analysis Example.^[15]

Correct: "John eats an apple."

↓ Parse Tree

```
S
 / \
NP  VP
 /  / \
John V  NP
   /  \
  eats DT N
      /  \
     an  apple
```

Incorrect: "Apple eats John an."

→ Grammatical error

Key Components.^[15]

1. **POS Tagging:** Assign noun/verb/adjective
2. **Ambiguity Resolution:** Handle multiple meanings
 - "I saw the man with the telescope"
 - Interpretation 1: I used telescope to see man
 - Interpretation 2: Man had telescope

Parse Tree Visualization:

```

S
/\
NP VP
/ \
John V NP
/ \
eats DT N
/ \
an apple

```

Semantic Processing

Definition: Understanding meaning of words and sentences. ^{[17][15]}

What It Ensures:

- Text is logically meaningful
- Contextually appropriate
- Not just grammatically correct

Semantic Analysis Tasks:^[15]

Task	Description	Example
Named Entity Recognition (NER)	Identify people, locations, organizations	"John lives in Gurugram" → John=Person, Gurugram=Location
Word Sense Disambiguation (WSD)	Determine correct meaning	"bank" = river vs money based on context

Semantic Example:^[15]

Sentence: "Apple eats a John."

↓

Grammatically: CORRECT ✓

Semantically: INCORRECT ✗ (apple cannot eat person)

Named Entity Recognition Example:^[15]

Text: "Elon Musk announced Tesla's new car in Austin, Texas."

NER Output:

Elon Musk → Person
Tesla → Organization
Austin → Location
Texas → Location

Word Sense Disambiguation Example:^[15]

Sentence 1: "I deposited money at the bank."

→ bank = financial institution ✓

Sentence 2: "We sat by the river bank."

→ bank = side of river ✓

Pragmatic Processing

Definition: Understanding **intended meaning** considering context, tone, and speaker intention.^[15]

What It Does:

- Interprets implied meanings
- Understands context and tone
- Identifies figurative language (idioms, metaphors)
- Supports conversational AI

Pragmatic Analysis Tasks:^[15]

Task	Description	Example
Understanding Intentions	Identify actual purpose	"Can you pass the salt?" → Request, not ability question
Figurative Meaning	Interpret non-literal	"I'm falling for you." → Love, not literal falling

Pragmatic Examples:^[15]

Statement	Literal Meaning	Pragmatic Meaning
"Can you pass the salt?"	Ability question	Polite request ✓
"I'm falling for you."	Physical falling	Expressing love ✓
"Hello! What time is it?"	Time inquiry	Concern about being late (context) ✓
"This is unfair!"	Statement of unfairness	Complaint/emotion (tone) ✓

Discourse Integration (Connecting sentences):^[15]

Text: "Taylor went to the store. She bought groceries."

↓

Anaphora Resolution: "She" → refers to "Taylor" ✓

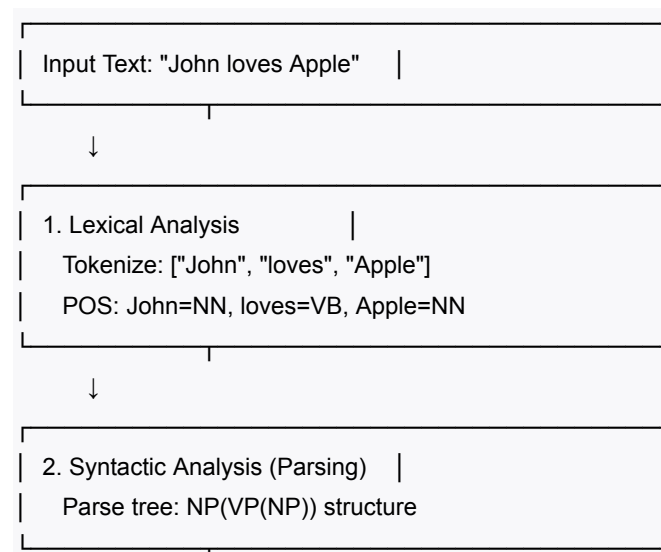
Text: "This is unfair!"

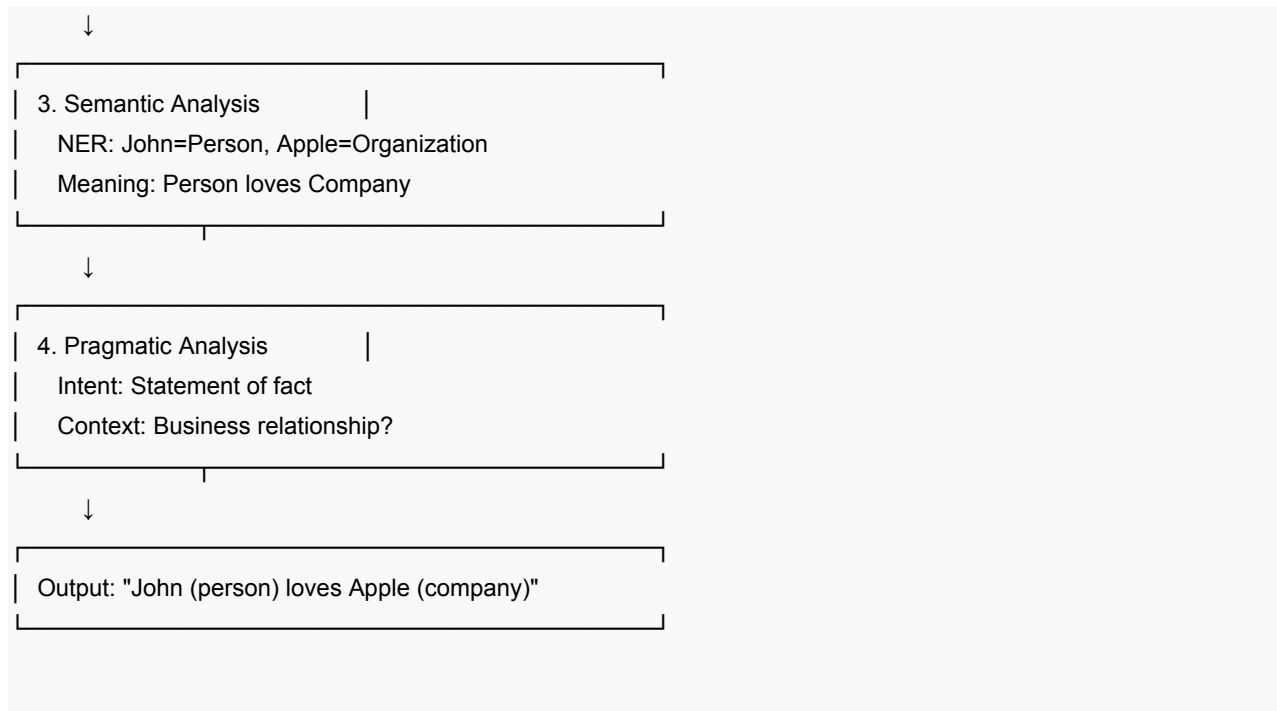
↓

Contextual Reference: "This" depends on surrounding context ✓

Complete NLP Pipeline

Full Processing Flow:^[15]





NLP vs Human Understanding:^[18]

Human: Understands context, sarcasm, emotion

NLP: Processes through 4 hierarchical layers

1. Lexical → words
2. Syntactic → grammar
3. Semantic → meaning
4. Pragmatic → intention

3. CODE IMPLEMENTATIONS

Python: Bayes' Theorem

```
def bayes_theorem(p_a, p_b_given_a, p_b_given_not_a):  
    """  
    Calculate P(A|B) using Bayes' Theorem  
    """  
    p_not_a = 1 - p_a  
    p_b = p_b_given_a * p_a + p_b_given_not_a * p_not_a  
    p_a_given_b = (p_b_given_a * p_a) / p_b  
    return p_a_given_b
```

```

# Spam Detection Example
p_spam = 0.3
p_offer_given_spam = 0.8
p_offer_given_not_spam = 0.1

result = bayes_theorem(p_spam, p_offer_given_spam, p_offer_given_not_spam)
print(f"P(spam|offer) = {result:.3f}") # Output: 0.774

```

Python: Fuzzy Logic Membership

```

def triangular_membership(x, a, b, c):
    """Triangular membership function"""
    if x <= a or x >= c:
        return 0.0
    elif a < x <= b:
        return (x - a) / (b - a)
    elif b < x < c:
        return (c - x) / (c - b)

# Temperature-based Fan Control
temp = 32
mu_warm = triangular_membership(temp, 20, 30, 40) # 0.8
mu_hot = triangular_membership(temp, 30, 40, 50) # 0.2

print(f"Temperature {temp}°C: Warm={mu_warm}, Hot={mu_hot}")

# Defuzzification (Centroid method)
medium_speed = 50
high_speed = 80
fan_speed = (mu_warm * medium_speed + mu_hot * high_speed) / (mu_warm + mu_hot)
print(f"Fan speed: {fan_speed:.1f}%") # Output: 56.0%

```

Python: NLP - POS Tagging

```

import nltk
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger')

from nltk import word_tokenize, pos_tag

sentence = "I am reading a book"

```

```

tokens = word_tokenize(sentence)
pos_tags = pos_tag(tokens)

print("Tokenization:", tokens)
print("POS Tags:", pos_tags)
# Output: [('I', 'PRP'), ('am', 'VBP'), ('reading', 'VBG'), ('a', 'DT'), ('book', 'NN')]

```

4. IMAGES & INFOGRAPHICS

[fuzzy_membership_types]

Membership Function Types: Shows triangular, trapezoidal, Gaussian, S-function shapes^[14]

[bayesian_network_example]

Bayesian Network: Medical diagnosis graph with Disease → Test_Result, Disease → Symptom^[11]

[nlp_pipeline]

NLP Processing Pipeline: Visual flow from lexical → syntactic → semantic → pragmatic analysis^[15]

[fuzzy_system_architecture]

Fuzzy Logic Architecture: Fuzzification → Rule Base → Inference → Defuzzification^[11]

5. REFERENCE SOURCES

Books (From Syllabus)

Book	Cod e	Chapters	Topics
Artificial Intelligence: A Modern Approach	TB1	Ch 7-8, 15	Uncertainty, Bayesian Networks, Fuzzy
Artificial Intelligence	TB2	Ch 13-14	Probabilistic Reasoning, NLP

Websites

Website	URL	Topics
GeeksforGeeks - Fuzzy Logic	fuzzy-logic ^[11]	Membership functions, fuzzy sets, applications
GeeksforGeeks - NLP Phases	nlp-phases ^[15]	Lexical, syntactic, semantic, pragmatic analysis
Towards Data Science - Bayes	bayes-theorem ^[3]	Bayes' theorem with spam & medical examples
Applied AI - Dempster-Shafer	dempster-shafer ^[8]	DST concepts, belief intervals, applications
Uncertainty Unit 3	uncertainty-pdf ^[1]	Complete uncertainty handling methods

YouTube Videos

Video	URL	Duration	Topics
Bayes Theorem in AI	bayes-video ^[19]	15 min	Bayes' theorem with applications & examples
NLP Analysis Layers	nlp-video ^[18]	20 min	Lexical, syntactic, semantic, pragmatic deep dive
Fuzzy Set Tutorial	fuzzy-video ^[12]	10 min	Fuzzy set definition, membership function
Dempster-Shafer Theory	Search: "Dempster Shafer AI"	15 min	DST concepts with murder case example

Additional Resources

- **Fuzzy Logic PDF:** [fuzzy-slides](#) - Types of membership functions ^[14]
- **Dempster-Shafer PDF:** [dst-slides](#) - MST, belief, plausibility ^[7]
- **NLP PDF:** [nlp-pdf](#) - Syntactic & semantic analysis ^[16]

- **Bayesian Network PDF:** [bayesian-ppt](#) - Bayes' theorem applications^[20]

Interactive Tools

- **Fuzzy Logic Calculator:** [fuzzy-calculator](#) - Design membership functions
 - **Bayesian Network Tool:** [bnmaker](#) - Build Bayesian networks
 - **NLP Playground:** [nlp-playground](#) - Test NLP models
 - **Python Fuzzy Library:** [fuzzylib](#) - Fuzzy logic in Python
-

QUICK RECAP: Key Formulas

Uncertainty Formulas

1. **Bayes' Theorem:** $P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$ ^{[3][4]}
2. **Probability Addition:** $P(A \vee B) = P(A) + P(B) - P(A \wedge B)$ ^[1]
3. **Certainty Factor:** $CF = \frac{P(H|E) - P(H)}{1 - P(H)}$ ^[1]
4. **Belief:** $Bel(A) = \sum_{B \subseteq A} m(B)$ ^[2]
5. **Plausibility:** $Pl(A) = 1 - Bel(\neg A)$ ^[2]

Fuzzy Logic Formulas

1. **Triangular Membership:**

$$\mu(x) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a < x \leq b \\ \frac{c-x}{c-b} & b < x \leq c \\ 0 & x \geq c \end{cases} \quad [1]$$

2. **Gaussian Membership:** $\mu(x) = e^{-\frac{(x-c)^2}{2\sigma^2}}$ ^[1]

3. **Centroid Defuzzification:** $Output = \frac{\sum \mu_i \cdot x_i}{\sum \mu_i}$ ^[1]

🎯 EXAM TIPS

Topic	What to Memorize
Bayes' Theorem	Formula + 2 examples (spam, medical)
Dempster-Shafer	Belief vs Plausibility + murder case example
Fuzzy Logic	Boolean vs Fuzzy table + triangular formula
Membership Functions	6 types + shapes (triangular, Gaussian, etc.)
Fuzzy Application	7 steps (universe → defuzzification)
NLP Phases	4 phases + 1 example each
Syntactic	Parse tree structure + POS tagging
Semantic	NER + WSD definitions
Pragmatic	Intention + figurative meaning

📊 SUMMARY TABLE: UNIT-III Coverage

Section	Topics	Hours	Key Concepts
Handling Uncertainty	Non-monotonic, probabilistic, Bayes, certainty factors, Bayesian networks, Dempster-Shafer, fuzzy logic	~6 hrs	Bayes formula, belief/plausibility, membership functions
NLP	Introduction, lexical, syntactic, semantic, pragmatic	~4 hrs	4 analysis phases, POS tagging, NER, WSD

Total: 10 hours ✓

Study Recommendation:

1. Start with uncertainty basics (probability, Bayes)
2. Practice Bayes' theorem problems (spam, medical)
3. Learn Dempster-Shafer belief/plausibility formulas
4. Master fuzzy logic (membership functions + design steps)
5. Study NLP phases sequentially (lexical → pragmatic)
6. Implement code examples (Bayes, fuzzy, POS tagging)
7. Review exam tips before test

Good luck! 🚀

*
**

-
1. <https://www.scribd.com/document/655465019/Unit-3-Uncertainty>
 2. <https://www.studocu.com/in/document/jawaharlal-nehru-technological-university-kakinada/artificial-intelligence/unit-iii-knowledge-representation-in-ai-concepts-techniques/138157775>
 3. <https://towardsdatascience.com/bayes-theorem-101-example-solution-ff54147d6c7f/>
 4. <https://www.geeksforgeeks.org/artificial-intelligence/bayes-theorem-in-artificial-intelligence/>
 5. <https://www.youtube.com/watch?v=RkJAKHlrUHc>
 6. <https://www.scribd.com/document/740378559/Bayesian-Theory-Bayesian-Network-Dempster-Shafer-Theory-AI-Seminar>
 7. <https://www.slideshare.net/slideshow/dempster-shafer-theory-ai-cse-8th-sem/89551632>
 8. <https://www.appliedaicourse.com/blog/dempster-shafer-theory-in-artificial-intelligence/>
 9. <https://www.geeksforgeeks.org/artificial-intelligence/heuristic-search-techniques-in-ai/>

10. <https://www.geeksforgeeks.org/artificial-intelligence/what-is-problems-problem-spaces-and-search-in-ai/>
11. <https://www.geeksforgeeks.org/artificial-intelligence/fuzzy-logic-introduction/>
12. <https://www.youtube.com/watch?v= evDAN0YUIE>
13. <https://www.youtube.com/watch?v=6h7Su5Yj30w>
14. <https://www.slideshare.net/slideshow/l4-types-of-membership-functions/49903433>
15. <https://www.geeksforgeeks.org/machine-learning/phases-of-natural-language-processing-nlp/>
16. <https://www.scribd.com/document/94796462/NLP>
17. <https://www.scribd.com/document/638610475/NLP-UNIT-2-Ques-ans-bank>
18. <https://www.youtube.com/watch?v=WaXOcZb9OhU>
19. <https://www.youtube.com/watch?v=oKko3ukFLVc>
20. <https://www.scribd.com/presentation/716888239/Unit-II-Probabilistic-Reasoning>
21. <https://arxiv.org/abs/1206.1319>
22. <https://www.scribd.com/document/886335585/Unit-3>
23. https://mrcet.com/downloads/digital_notes/CSE/III Year/CS/Natural Language Processing-1.pdf
24. <https://www.geeksforgeeks.org/artificial-intelligence/representing-knowledge-in-an-uncertain-domain-in-ai/>