

## NLP – Unit 6 (Applications of NLP) – END-SEM PYQ Answers

May–June 2023

### Q7(a) Sentiment Analysis

[10 Marks]

- **Definition:** Sentiment Analysis (Opinion Mining) is the computational study of people's opinions, sentiments, emotions, and attitudes expressed in natural language text toward entities such as products, services, events, topics, or organisations.

- **Levels of Sentiment Analysis:**

Level	Granularity	Description	Example
Document-level	Whole document	Overall polarity of the entire review/article	A 5-star Amazon review → POSITIVE
Sentence-level	Per sentence	Sentiment of each individual sentence	'I loved the plot. The acting was terrible.'
Aspect-level (ABSA)	Per feature	Sentiment toward specific product attributes	camera: +ve, battery: -ve, price: neutral
Emotion Detection	Fine-grained	Specific emotion beyond pos/neg/neutral	'I was shocked!' → SURPRISE

- **Four Approaches to Sentiment Analysis:**

Approach	Mechanism	Pros	Cons
Lexicon-Based	Count +/- words from predefined dictionary; sum scores	No training data; transparent; fast	Cannot handle negation ('not bad'), sarcasm, domain shift
Classical ML	Train Naïve Bayes, SVM on TF-IDF features	Learns domain-specific patterns; portable	Needs labelled data; feature engineering effort
Deep Learning	CNN, LSTM, BiLSTM on word embeddings	Captures sequential context	Computationally heavier; more data needed
Transformer-Based	Fine-tune BERT, RoBERTa, DistilBERT for sentiment	Highest accuracy; handles context, negation, sarcasm	GPU required; slow inference

- **Sentiment Analysis Pipeline:**

```

INPUT TEXT
↓
PREPROCESSING
• Tokenise, lowercase, remove URLs/usernames
• Handle negation: 'not good' → append NOT_ prefix: 'not_good'
• Remove noisy characters, expand contractions
↓
FEATURE EXTRACTION
• Bag of Words (BOW) / TF-IDF vectors
• Word embeddings (Word2Vec, GloVe, FastText)
• BERT contextual embeddings
↓
MODEL PREDICTION
• Lexicon: sum sentiment scores
• ML: predict class from features
• Transformer: [CLS] representation → linear classifier
↓
OUTPUT: Sentiment Label + Confidence Score

```

- **Handling Sentiment Challenges:**

Challenge	Example	Solution
Negation	'Not bad at all' is positive	NOT_ prefix propagation; negation scope detection
Sarcasm	'Oh great, another bug!' is negative	Tone markers, context, multimodal cues
Aspect ambiguity	'Small screen but great camera'	Aspect-level SA (ABSA)
Domain shift	'Unpredictable' = good in thriller, bad in car review	Domain-adapted training; domain-specific lexicon

*Aspect-Based Sentiment Analysis (ABSA) is the most practically useful form: instead of an overall 4/5 rating, ABSA gives camera: excellent, battery: poor, design: good. BERT fine-tuned for ABSA achieves 85%+ accuracy. VADER is specifically designed for social media — handles punctuation emphasis ('GREAT!!!'), emoticons, slang, and degree modifiers ('very', 'slightly').*

## Q7(b) Statistical Machine Translation (SMT)

[7 Marks]

- **SMT Core — Noisy Channel Model:**

Goal: Find best English translation  $e^*$  for French sentence  $f$

$$\begin{aligned}
 e^* &= \operatorname{argmax}_e P(e | f) \\
 &= \operatorname{argmax}_e [P(f | e) \times P(e)] \quad [\text{Bayes' Theorem}]
 \end{aligned}$$

$P(f | e)$  = Translation Model (TM): how likely is French  $f$  given English  $e$ ?

$P(e)$  = Language Model (LM): how fluent/natural is English  $e$ ?

- **Three Core SMT Components:**

Component	Role	Training Data
Translation Model $P(f e)$	Maps source phrases to target phrase probabilities	Bilingual parallel corpus (aligned sentence pairs)
Language Model $P(e)$	Ensures target text is fluent and grammatical	Large monolingual target corpus
Decoder	Finds $e^* = \operatorname{argmax} P(f e) \times P(e)$ efficiently via beam search	No additional training needed

- **SMT Training Pipeline (8 Steps):**

- Step 1 – Collect parallel corpus (millions of sentence pairs): Europarl, UN Corpus, OPUS
- Step 2 – Sentence alignment: Gale–Church algorithm matches source–target sentence pairs
- Step 3 – Word alignment: IBM Models 1–5 via GIZA++ — assigns each source word to target word(s)
- Step 4 – Phrase extraction: extract consistent phrase pairs from word alignments
- Step 5 – Build phrase table:  $P(\text{target\_phrase} | \text{source\_phrase})$  from phrase pair frequencies
- Step 6 – Train N-gram LM on large target monolingual corpus (KenLM, SRILM)
- Step 7 – MERT tuning: tune weights of TM, LM, and reordering model on development set
- Step 8 – Decode: Moses decoder applies beam search + phrase table + LM to produce translation

- **BLEU Score (MT Evaluation Metric):**

$$\text{BLEU} = \text{BP} \times \exp(\sum_n [w_n \times \log(p_n)])$$

BP = brevity penalty (penalises too-short translations)

$p_n$  = n-gram precision: fraction of output n-grams found in reference

$w_n$  = weight (usually 1/4 for  $n = 1, 2, 3, 4$ )

BLEU 1.0 = perfect match; 0.0 = no overlap

Commercial MT systems: BLEU  $\approx 0.40$ – $0.60$

*Google Translate switched from SMT to NMT in November 2016 — a single switch that improved translation quality more than 10 years of SMT improvements. However, SMT concepts (phrase tables, LM scoring, beam search, BLEU) remain foundational. The phrase table in SMT plays the same conceptual role as the attention mechanism in NMT: both align source with target.*

**Q8(a) Machine Translation Approaches****[10 Marks]**

- 1. Rule-Based Machine Translation (RBMT):**

Sub-type	Approach	Example
Direct	Word-for-word dictionary substitution; minimal linguistic analysis	French 'chat' → English 'cat' (dictionary lookup)
Transfer	Full syntactic analysis of source; apply transfer grammar; generate target	French SVO parse → transfer grammar → English SVO parse
Interlingua	Convert source to language-independent semantic representation; generate any target	French → [semantic form] → Japanese

- RBMT Transfer Process — Detailed:**

Input: 'Il mange une pomme' (French: He eats an apple)

Step 1 — Morphological Analysis (French):

Il = PRON(3rd, masc, sing, subject)  
 mange= VERB(manger, indicative, present, 3sg)  
 une = DET(indefinite, feminine, singular)  
 pomme= NOUN(feminine, singular)

Step 2 — Syntactic Parse (French):

[S [NP Il/PRON] [VP mange/VERB [NP une/DET pomme/NOUN]]]

Step 3 — Transfer Rules:

Il (3sg masc subj pron) → 'He'  
 manger.3sg.pres → 'eats'  
 une (indef art) → 'an'  
 pomme (fruit noun) → 'apple'

Step 4 — Morphological Generation (English):

Apply English grammar: subject He → verb eats (3sg agreement)

Step 5 — Output: 'He eats an apple'

- 2. SMT:**

**REPEATED — Refer to: May–June 2023 → Q7(b) [Full SMT noisy channel model, 3 components, 8-step pipeline, BLEU]**

- 3. Neural Machine Translation (NMT):**

- Architecture:** Encoder–Decoder with Attention Mechanism (Bahdanau 2014). The Transformer ('Attention Is All You Need', Vaswani 2017) is the current standard.

Encoder: reads entire source sentence → sequence of context vectors (one per source token)

Attention mechanism at decoder step  $t$ :

$e(t,j)$  = score(decoder\_hidden <sub>$t$</sub> , encoder\_hidden <sub>$j$</sub> ) [compatibility]  
 $a(t,j)$  = softmax( $e(t,j)$ ) over all  $j$  [normalize]  
 context <sub>$t$</sub>  =  $\sum_j a(t,j) \times \text{encoder\_hidden}_j$  [weighted sum]

Decoder: generates target tokens one by one, conditioned on attention-weighted context

Feature	RBMT	SMT	NMT
Knowledge source	Hand-crafted linguistic rules	Statistical patterns from parallel corpus	End-to-end neural network
Training data	None (rules written manually)	Millions of sentence pairs	Hundreds of millions of sentence pairs
Translation quality	Medium — misses idioms, informal text	Good — captures statistical patterns	Excellent — near-human on many pairs
Fluency	Low — often unnatural output	Medium — phrase stitching artifacts	High — smooth, natural-sounding
Interpretable?	Yes — every decision traceable	Partially (phrase tables readable)	No — black box neural network
Long-range dependencies	No	Partially	Yes — attention spans full sentence
Production examples	Systran (1970s–2000s)	Moses (2006–2016)	Google Translate (2016+), DeepL, Microsoft

*The Transformer architecture (2017) eliminated the need for RNNs in NMT. Multi-head self-attention enables full parallelism during training (unlike LSTM which must process sequentially). The Transformer is also the foundation for BERT, GPT, T5, and all modern large language models.*

### Q8(b) Natural Language Generation (NLG) with Reference Architecture [7 Marks]

- **NLG Definition:** The task of automatically producing natural language text from non-linguistic input such as structured data, knowledge graphs, database records, or semantic representations.

System	Input	Output	Function
NLG	Structured data, facts, database records	Natural language text (sentences, paragraphs)	Generate text FROM meaning/data
NLU	Natural language text	Structured representation (intent, slots, facts)	Understand meaning FROM text
TTS	Natural language text (string)	Spoken audio waveform	Convert text to speech (not NLG)
Full Pipeline	Raw data	Spoken natural language	NLG → text → TTS → speech

- **NLG Reference Architecture — Reiter and Dale (1997) Three Stages:**
- **STAGE 1 — Document Planning (WHAT to say and in what order):**
  - Content Determination: select relevant facts/data from the input; filter out irrelevant data
  - Document Structuring: decide the order and rhetorical organisation (contrast, elaboration, sequence)

- **STAGE 2 — Micro-Planning / Sentence Planning (HOW to say it):**
  - Aggregation: group related facts into single coherent sentences
  - Lexicalisation: choose specific words to express concepts ('precipitation' vs 'rain' vs 'drizzle')
  - Referring Expression Generation (REG): decide how to refer to entities to avoid repetition
- **STAGE 3 — Surface Realisation (produce grammatical text):**
  - Linguistic realisation: apply grammar rules — subject–verb agreement, tense consistency, determiners
  - Orthographic realisation: handle capitalisation, punctuation, paragraph breaks, formatting
- **NLG Pipeline Example — Weather Report:**

INPUT: temperature=32°C, condition=sunny, wind=15 km/h NE, UV=high

STAGE 1 (Content Selection):

Include: temperature + condition + wind + UV

Order: current conditions → advisory

STAGE 2 (Sentence Planning):

Aggregate: temperature + condition → single sentence

Lexicalise: 32°C = 'hot'; 15 km/h NE = 'light northeasterly breeze'

STAGE 3 (Realisation):

'Today will be hot and sunny with temperatures reaching 32 degrees.'

'A light northeasterly breeze is expected throughout the day.'

'UV levels will be high — sunscreen is recommended.'

*Modern NLG has largely moved to end-to-end neural approaches (T5, GPT-4). However, the Reiter–Dale (1997) architecture remains the standard conceptual framework and is still used for rule-based/template NLG in regulated domains (medical, legal, financial) where controllability and explainability are required.*

## November–December 2023

### Q7(a) Question Answering System: Three Stages

[7 Marks]

- **QA System:** Automatically answers natural language questions by finding or generating relevant answers from a text corpus or knowledge base.
- **Types of QA Systems:**

Type	Description	Examples
Factoid QA	Short factual answers (named entity, number, date)	Who is the CEO of Apple? → Tim Cook
Open-domain QA	Answers from a large corpus (e.g., Wikipedia)	IBM Watson, Google snippet answers
Machine Reading Comprehension	Answer span extracted from a given passage	SQuAD benchmark
Generative QA	Answer generated rather than extracted	ChatGPT, Google Gemini, Perplexity AI

- **STAGE 1 — Question Processing / Analysis:**

- Question Type Classification: WHO → PERSON, WHAT → DEFINITION, WHEN → DATE, WHERE → LOCATION, HOW MUCH → QUANTITY
- Expected Answer Type Detection: guides what kind of entities to look for in passages
- Keyword/Query Term Extraction: identify terms to use in document retrieval

Example: 'When did India gain independence from British rule?'

Type: WHEN

Expected Answer Type: DATE

Keywords: India, independence, British, rule

- **STAGE 2 — Document / Passage Retrieval:**

- Sparse retrieval: BM25 / TF-IDF inverted index — fast, no GPU needed
- Dense retrieval: DPR (Dense Passage Retrieval) — BERT bi-encoder; semantic matching
- Re-ranking: cross-encoder BERT reads query + passage together for fine-grained relevance scoring

- **STAGE 3 — Answer Extraction / Generation:**

- Span extraction (factoid QA): fine-tuned BERT predicts start and end token positions of answer span within passage
- Generative answer: seq2seq model (T5, GPT) generates fluent answer conditioned on question + retrieved passages
- RAG (Retrieval-Augmented Generation): combines retrieval (Stage 2) + generation (Stage 3) — current best approach

Passage: 'India gained independence on 15 August 1947, ending nearly 200 years of British colonial rule.'

BERT-QA:

Question: 'When did India gain independence?'

Expected type: DATE

Answer span: '15 August 1947'

*Modern production QA (Google Featured Snippets, Alexa, Siri) uses RAG: Stage 2 retrieves with BM25+DPR, Stage 3 generates with a large language model (GPT-4, Gemini) conditioned on retrieved passages. BERT achieves 85%+ EM on SQuAD 1.1, surpassing human performance of 82.3%.*

## Q7(b) Rule-Based MT and Statistical MT

[10 Marks]

**REPEATED — Refer to: May–June 2023 → Q8(a) [RBMT 3 sub-types + detailed transfer example] + Q7(b) [SMT noisy channel, components, pipeline, BLEU]**

**Q8(a) NLP Applications: Text Entailment and Dialog Agents****[10 Marks]**

- **i) Text Entailment (Natural Language Inference — NLI):**
- **Definition:** Given a PREMISE and a HYPOTHESIS, determine: does the premise ENTAIL the hypothesis, CONTRADICT it, or is the relationship NEUTRAL?

Label	Meaning	Premise	Hypothesis
Entailment	If premise is true, hypothesis must be true	A black dog is running through snow.	An animal is outdoors.
Contradiction	Premise makes hypothesis impossible	The man is sleeping on a park bench.	The man is exercising.
Neutral	No logical connection	A woman is playing piano in the hall.	The woman is a professional musician.

- **NLI Applications:**
  - Fact checking: Does news article A support or contradict claim B?
  - QA verification: Does the retrieved passage entail the candidate answer?
  - Summarisation evaluation: Does the summary preserve the meaning of the source?

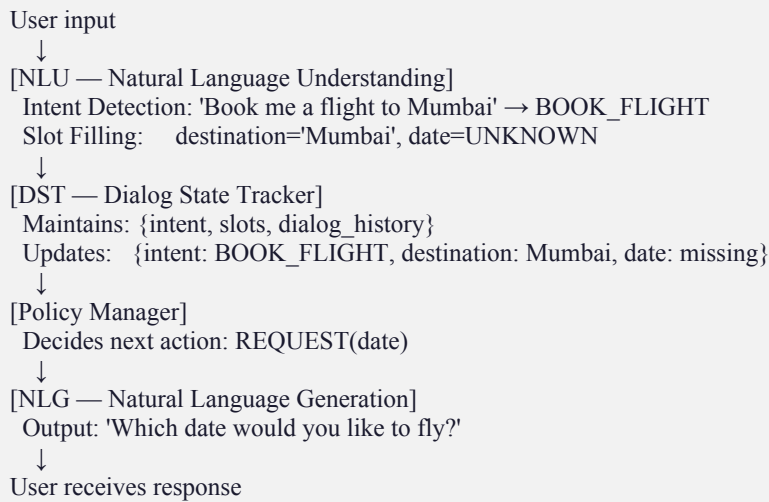
Relation	Symmetric?	Example	Key Property
Entailment	No (one-way)	'dog running' → 'animal moving'	If P is true, H must be true
Contradiction	Yes (symmetric)	'sleeping' contradicts 'running'	If P is true, H must be false
Paraphrase	Yes (symmetric)	'He left early' ≡ 'He departed before time'	Bidirectional entailment — same meaning
Semantic Similarity	Yes (0–1 scale)	$\text{sim}(\text{car}, \text{bus}) = 0.7$	Graded, not binary

- **ii) Dialog and Conversational Agents:**

Type	Goal	Example
Task-oriented	Complete a specific task efficiently	'Set alarm for 7 AM' → alarm_time = 7AM
Open-domain (chitchat)	Natural free-form conversation	Alexa small talk, ChatGPT casual conversation
QA-based	Answer user questions from a knowledge source	Customer support bot, FAQ assistant
Hybrid	Handle both tasks and open-domain conversation	Siri, Google Assistant, Cortana

- **Modular Dialog System Architecture:**





*Modern conversational AI (ChatGPT, Claude, Gemini) replaces the explicit modular architecture with a single large language model that handles all steps implicitly. The modular architecture is still used for production task-oriented systems (RASA, Dialogflow CX, Amazon Lex) because it is more controllable, auditable, and can enforce business rules. RLHF (Reinforcement Learning from Human Feedback) is used to align LLM-based dialogue systems with human preferences.*

### Q8(b) Natural Language Generation

[7 Marks]

**REPEATED — Refer to: May–June 2023 → Q8(b) [NLG full 3-stage Reiter–Dale architecture with weather example and applications]**

### May–June 2024 [6263]-96

### Q7(a) NLG: Role, Difference from TTS, Applications

[9 Marks]

**REPEATED — Refer to: May–June 2023 → Q8(b) [NLG 3-stage architecture, NLG vs NLU vs TTS table, applications]**

- **Extended — NLG Evaluation Metrics:**

Metric	What It Measures	Limitation
BLEU	N-gram precision vs human references	High BLEU ≠ human quality; ignores semantics
ROUGE	N-gram recall (summarisation)	Designed for summarisation; not general NLG
METEOR	F-score with stemming + synonym matching (WordNet)	Better than BLEU but slower
BERTScore	Semantic similarity using BERT embeddings (cosine of token embeddings)	Better semantic quality estimate
Human evaluation	Fluency, accuracy, relevance ratings (1–5 scale)	Expensive, time-consuming, subjective

**Q7(b) Cross-Lingual Translation Challenges****[8 Marks]**

Challenge	Description	Real Example	Solution
Word Order	SOV vs SVO vs VSO structures	English: 'John eats rice' / Japanese: 'John rice eats'	Transformer attention handles reordering implicitly
Morphological Richness	Agglutinative languages pack many meanings per word	Finnish 'talossani' = 'in my house' (1 word = 4 English)	Subword tokenisation (BPE, SentencePiece)
Lexical Gaps	Source concept has no direct target equivalent	Portuguese 'saudade' — no English word	Approximate translation + explanation; borrow the word
Idiomatic Expressions	Literal translation is nonsensical	'Kick the bucket' → 'die' (not literally)	Idiom detection; NMT learns idioms from data
Grammatical Gender	Source lacks gender; target requires it	English 'doctor' → French 'docteur'(m) or 'doctoresse'(f)?	Context-based gender inference
Low-resource Pairs	Limited parallel training data	Swahili–Czech, Hindi–Odia	Cross-lingual transfer, back-translation, zero-shot NMT
Code-switching Queries	Query mixes two languages	'Best restaurants near mujhe' (English+Hindi)	Multilingual BERT; language detection first

**Q8(a) RBMT vs Statistical MT****[9 Marks]**

**REPEATED — Refer to: May–June 2023 → Q8(a) [RBMT sub-types + transfer example] + Q7(b) [SMT full pipeline]**

Dimension	RBMT	SMT
Knowledge encoding	Explicit linguistic rules (grammar, lexicon)	Implicit — statistical patterns from data
Linguistic analysis depth	Deep (morphology + syntax + semantics)	Shallow (phrase alignment, no full parse)
Error types	Rule gaps → systematic, predictable failures	Data sparsity → inconsistent, hard to predict
Domain adaptation	Write new domain rules (very expensive)	Collect domain-parallel corpus + retrain (cheaper)
Ambiguity resolution	Explicit disambiguation rules or heuristics	LM selects statistically most likely translation
Maintenance	Update rules — transparent but laborious	Collect more data + retrain (more scalable)

**Q8(b) Conversational Agent: NLG and NLU Components****[8 Marks]**

**REPEATED — Refer to: Nov–Dec 2023 → Q8(a) ii [Full dialog architecture code + NLU table + NLG methods]**

- **NLU Tasks in Dialog:**

NLU Task	Description	Example	Tool
Intent Detection	Classify utterance into predefined intent	'Set alarm for 7 AM' → ALARM_SET	BERT fine-tuned on intent dataset
Slot Filling	Extract parameter values for the intent	alarm_time = '7 AM'	Span prediction; sequence labelling
Coreference in Dialog	Resolve pronouns across turns	'Book it for tomorrow.' → 'it' = flight	SpanBERT coreference model
Dialog Act Classification	Label speech act type	'Could you book a flight?' = REQUEST	BERT + dialog act taxonomy

- **NLG Methods in Dialog:**

NLG Method	Mechanism	Use Case
Template-Based	Fill slots in pre-written templates	Customer service, FAQ bots
Retrieval-Based	Select best pre-written response from database	Chitchat, support bots
Generative (seq2seq)	Encode dialog state; decode response	Open-domain dialogue
LLM-Based (GPT/Claude)	Condition on dialog history; generate freely	Modern AI assistants
Hybrid	Use templates/retrieval by default; LLM for novel cases	Production systems

## May–June 2025 [6404]-96

### Q7(a) RBMT vs SMT Architecture

[9 Marks]

**REPEATED — Refer to: May–June 2023 → Q8(a) [RBMT] + Q7(b) [SMT] + 3-way RBMT/SMT/NMT comparison table**

### Q7(b) QA System: Components and ML/NLP Techniques

[8 Marks]

**REPEATED — Refer to: Nov–Dec 2023 → Q7(a) [QA 3 stages + types table]**

- **ML Techniques per QA Stage:**

QA Stage	Classical ML	Deep Learning	Transformer-Based
Question Processing	MaxEnt classifier for question type	BiLSTM for type classification	BERT classification on [CLS] token
Document Retrieval	TF-IDF, BM25, inverted index	CNN-based passage scoring	DPR (Dense Passage Retrieval) bi-encoder
Answer Extraction	Pattern matching, NER-based	BiDAF (Bidirectional Attention Flow)	BERT-QA: predict start/end token positions
Answer Generation	Template selection	seq2seq LSTM decoder	T5/GPT-4 RAG (Retrieval-Augmented Generation)

- **RAG — Modern QA Architecture:**

Step 1: User asks: 'Who invented the telephone?'

Step 2: Dense Retrieval (DPR)

Query encoder → query vector

FAISS index search over 21M Wikipedia passage vectors

Retrieve top-k = 5 most relevant passages

Step 3: Generation (GPT-4 / T5)

Input: 'Based on these passages: [p1] [p2] ...

Answer: Who invented the telephone?'

Output: 'Alexander Graham Bell is credited with inventing the telephone in 1876.'

Advantages:

- Factual grounding — answer comes from retrieved evidence, not hallucination
- Updateable: swap in newer documents without retraining the LLM

### Q8(a) Text Entailment: Importance and Differences

[9 Marks]

**REPEATED — Refer to: Nov–Dec 2023 → Q8(a) i [Text entailment 3-label table + NLI applications + comparison table]**

- **Why Entailment ≠ Similarity:**

Example 1 — Similar but NOT entailment:

'A dog is barking' vs 'A cat is meowing'

Similarity: both animals making sounds → moderate similarity score

Entailment: NO — dog barking does NOT imply cat meowing

Example 2 — Entailment (one-way):

'The man is jogging' ENTAILS 'A person is moving'

BUT: 'A person is moving' does NOT entail 'The man is jogging'

Entailment is DIRECTIONAL; similarity is SYMMETRIC

Example 3 — Paraphrase = bidirectional entailment:

'He left early' ⇔ 'He departed before the scheduled time'

A ⇒ B and B ⇒ A → PARAPHRASE

- **NLI Datasets and Benchmarks:**

Dataset	Size	Description	BERT Performance
SNLI (Stanford NLI)	570K pairs	Image captions as premises; hypotheses crowd-sourced	~91% accuracy
MultiNLI	433K pairs	10 diverse genres (news, fiction, government)	~86% accuracy
ANLI (Adversarial NLI)	162K pairs	3 rounds of adversarially collected hard examples	~50% (much harder)
FeverousNLI	185K pairs	Claims from Wikipedia with evidence	~85% accuracy

### Q8(b) NLG with Reference Architecture

[8 Marks]

**REPEATED — Refer to: May–June 2023 → Q8(b) [Full NLG 3-stage Reiter–Dale architecture with weather example]**

## November–December 2025

### Q7(a) RBMT vs SMT Architecture Comparison

[9 Marks]

**REPEATED — Refer to: May–June 2023 → Q8(a) [RBMT types + transfer example] + Q7(b) [SMT pipeline]**

Pipeline Stage	RBMT	SMT
Input	Raw source sentence	Raw source sentence
Source Analysis	Full syntactic parse + morphological analysis	Word alignment with IBM models (GIZA++)
Core Mapping	Transfer grammar rules (handcrafted)	Phrase table lookup: $P(\text{target} \mid \text{source phrase})$
Language Model	Grammar generation rules ensure correctness	N-gram LM: $P(e)$ penalises ungrammatical output
Output Selection	Deterministic — rules determine unique output	Beam search: $\text{argmax } P(f e) \times P(e)$
Training cost	Years of linguist effort	Weeks of compute on parallel corpus
Maintenance	Edit rules (transparent but laborious)	Collect more data + retrain (scalable)

### Q7(b) NLG Reference Architecture

[8 Marks]

**REPEATED — Refer to: May–June 2023 → Q8(b) [Full NLG 3-stage with weather example + evaluation metrics table]**

### Q8(a) Question Answering: Architecture and Challenges

[9 Marks]

**REPEATED — Refer to: Nov–Dec 2023 → Q7(a) [QA types + 3-stage pipeline] + May–June 2025 → Q7(b) [ML techniques table + RAG code]**

- QA System Challenges:**

Challenge	Description	Research Direction
Multi-hop Reasoning	Answer requires connecting facts across multiple passages	HotpotQA; multi-hop transformers
Unanswerable Questions	Corpus doesn't contain the answer; system must detect this	SQuAD 2.0; calibrated confidence scores
Long-context QA	Relevant answer buried in 100-page document	Longformer, BigBird, sliding window BERT
Numerical Reasoning	QA requires arithmetic or counting	NumNet, MathQA, program synthesis
Table QA	Answer in a structured table, not prose	TAPAS (Google), SQL generation models
Multilingual QA	Question and document in different languages	mBERT fine-tuned on MKQA, XQuAD

**Q8(b) Conversational Agent Components****[8 Marks]****REPEATED — Refer to: Nov–Dec 2023 → Q8(a) ii + May–June 2024 → Q8(b) [Full dialog architecture code + NLU table + NLG methods]**

- **Modern LLM-Based Dialogue:**
  - RLHF (Reinforcement Learning from Human Feedback): GPT pre-training → Supervised Fine-Tuning (SFT) → Reward Model training → PPO optimisation against reward model
  - Constitutional AI: model fine-tuned against a set of principles (constitution) without direct human feedback at every step
  - RAG-based chatbots: retrieve relevant knowledge base passages at query time; generate response grounded in retrieved evidence; avoids hallucination
  - Evaluation: perplexity (fluency), BLEU/ROUGE (reference-based), human ratings (engagement, accuracy, helpfulness)

**Topic Frequency Analysis — Unit 6**

Rank	Topic	Sessions	Priority
1	NLG (Natural Language Generation) — 3-stage architecture	MJ23, ND23, MJ24, MJ25, ND25	VERY HIGH — 5× every exam
2	RBMT vs SMT — all approaches + NMT comparison	MJ23, ND23, MJ24, MJ25, ND25	VERY HIGH — 5× every exam
3	Question Answering Systems (3 stages)	ND23, MJ24, MJ25, ND25	HIGH — 4×
4	Dialog and Conversational Agents (full architecture)	ND23, MJ24, MJ25, ND25	HIGH — 4×
5	Sentiment Analysis (all approaches)	MJ23	MEDIUM — 1× (10-mark standalone)
6	Text Entailment / NLI	ND23, MJ25	MEDIUM — 2×
7	Cross-Lingual Translation Challenges	MJ24	MEDIUM — 1×

**EXAM STRATEGY — Unit 6:** (1) NLG: ALWAYS write all 3 stages with the Reiter–Dale (1997) attribution. Use the weather report pipeline example. (2) RBMT: Know the 5-step transfer process (morphological analysis → syntactic parse → transfer rules → morphological generation → output) with the French–English example. (3) SMT: Know the noisy channel formula  $e^* = \operatorname{argmax} P(f|e) \times P(e)$ , the 3 components, and BLEU score. (4) QA: Know all 3 stages with BERT-QA span extraction. Write RAG as the modern approach for bonus marks. (5) Dialog: Draw the full pipeline (NLU → DST → Policy → NLG) with an example for each stage.