Print / Save as PDF

Click button, then choose "Save as PDF"

NLP MIDSEM

Super Simple Prep Guide

Explained Like You're 5 Years Old

Score 20+ in 5 Hours

📋 Table of Contents

🗺️ Exam Map - Where Marks Are Hiding **Page 3**

⏰ Your 5-Hour Battle Plan **Page 4**

📌 Master Formula Card **Page 5**

📗 Hour 1: TF-IDF & Cosine Similarity **Page 9**

📘 Hour 1 (Cont.): Word Embeddings **Page 12**

📙 Hour 2: HMM & Viterbi Algorithm **Page 15**

📕 Hour 3: Language Models **Page 20**

📒 Hour 4: Theory Questions **Page 23**

⚠️ Common Mistakes **Page 25**

🏆 Exam Strategy **Page 26**

🗺️ EXAM MAP - Where Marks Are Hiding

Q#	What They'll Ask	Type	Marks	Difficulty
1️⃣	Introduction - NLP Apps	Write sentences	4	😊 Easy
2️⃣	Language Models	🔢 Calculate	4	😐 Medium
3️⃣	Neural LM & LLM	Write + Apply	4	😊 Easy
4️⃣	Vector Semantics	🔢 Calculate	4	😐 Medium
5️⃣	Word Embeddings	🔢 Calculate	5	😐 Medium
6️⃣	POS Tagging	🔢 Calculate	4	😐 Medium
7️⃣	Viterbi Algorithm	🔢 Calculate	5	😓 Hard

> **🎯 Secret: 26 out of 30 marks = JUST CALCULATIONS. Learn formulas = Win!**

⏰ YOUR 5-HOUR BATTLE PLAN

Time	What to Study	Expected Marks
11:45 AM - 12:45 PM	Q4 + Q5 (TF-IDF, Cosine, Word2Vec)	**+9**
12:45 PM - 1:45 PM	Q6 + Q7 (HMM, Viterbi)	**+9**
1:45 PM - 2:45 PM	Q2 + Q3 (N-gram, Perplexity, LLM)	**+8**
2:45 PM - 3:45 PM	Q1 (Theory + All Formula Review)	**+4**
3:45 PM - 4:45 PM	Practice 5 problems, eat, relax	🧘

📌 MASTER FORMULA CARD (Screenshot This!)

🧮 FORMULA 1: TF-IDF

$$TF = 1 + \log_{10}(count)$$
$$IDF = \log_{10}(N \div df)$$
$$\mathbf{TF\text{-}IDF = TF \times IDF}$$

count = times word appears in document

N = total documents

df = documents with word

⏰ **When to Use:**
Q4 asks "Calculate TF-IDF for word X in document Y"

🔢 **Quick Examples:**

Example 1: count=5, N=500, df=100
  TF = 1 + log(5) = 1.7
  IDF = log(500÷100) = 0.7
  TF-IDF = 1.7 × 0.7 = 1.19 ✓

🧮 FORMULA 2: COSINE SIMILARITY

$$Cosine = (a_1 \times b_1 + a_2 \times b_2 + \ldots) \div (\sqrt{(a_1^2 + a_2^2 + \ldots)} \times \sqrt{(b_1^2 + b_2^2 + \ldots)})$$

⏰ **When to Use:**
Q4 asks "Find similarity between two vectors"

🔢 **Quick Example:**

```
A=[3,4], B=[4,3]
  Dot = 3×4 + 4×3 = 24
  Length A = √(9+16) = 5
  Length B = √(16+9) = 5
  Cosine = 24÷25 = 0.96 ✔
```

🧮 FORMULA 3: PERPLEXITY

PP = (1 ÷ P)^(1/N)

P = multiply all word probabilities
N = number of words

💡 **LOWER = BETTER!**

⏰ **When to Use:**
Q2 asks "Calculate perplexity for this sentence"

🔢 **Quick Example:**

P(I)=0.4, P(love|I)=0.5, P(NLP|love)=0.2, N=3
P = 0.4 × 0.5 × 0.2 = 0.04
PP = (1÷0.04)^(1/3) = 2.92 ✔

🧮 FORMULA 4: BIGRAM & LAPLACE

BIGRAM:

P(word|prev) = Count(prev,word) ÷ Count(prev)

LAPLACE:

P(word|prev) = (Count + 1) ÷ (Count(prev) + V)

Use when Count = 0!

🔢 **Quick Examples:**

Bigram: C(I,love)=2, C(I)=2
P(love|I) = 2 ÷ 2 = 1.0 ✔

Laplace: C(the,cat)=0, C(the)=50, V=10000
P(cat|the) = 1÷10050 = 0.0001 ✔

🧮 FORMULA 5: HMM SCORE

Score(tag) = P(tag|prev_tag) × P(word|tag)

TRANSITION × EMISSION

🎯 **Pick tag with HIGHEST score!**

⏰ **When to Use:**
Q6 asks "Which tag should this word get?"

🔢 **Quick Example:**

```
Word "flies" after Noun:
  Try NN: 0.3 × 0.02 = 0.006
  Try VBZ: 0.4 × 0.05 = 0.020 ← Winner! ✓
```

🧮 FORMULA 6: WORD2VEC UPDATE

$$Error = \sigma(v \cdot u) - y$$

(y=1 for real, y=0 for fake)

$$v_{new} = v_{old} - \eta \times Error \times u$$

💡 Real pair → vectors move CLOSER
💡 Fake pair → vectors move APART

🔢 **Quick Example:**

```
REAL pair (cat, meow), y=1
  σ(v·u) = 0.55
```

```
Error = 0.55 − 1 = −0.45
```

v moves TOWARD u ✔

🧮 FORMULA 7: WORD ANALOGY

v_? = v_known − v_old_context + v_new_context

Pattern: A is to B as C is to ?

Formula: ? = C − A + B

⏰ **When to Use:**
Q5 asks "Find the vector using analogy"

🔢 **Quick Examples:**

King:Man :: Queen:Woman

Queen = King − Man + Woman ✓

v_Man=[0.5,0.3], v_Woman=[0.4,0.6], v_King=[0.8,0.4]
v_Queen = [0.8,0.4] − [0.5,0.3] + [0.4,0.6]
= [0.7, 0.7] ✓

🧮 FORMULA 8: VITERBI

INIT:

$V_1(tag) = \pi(tag) \times P(word_1|tag)$

RECURSE:

$V_t(tag) = \max[V_{t-1}(prev) \times P(tag|prev)] \times P(word_t|tag)$

BACKTRACK:

Start from max final, follow pointers back

⏰ **When to Use:**

Q7 asks "Find best tag sequence" or "Complete Viterbi table"

⚠️ COMMON MISTAKES - DON'T DO THESE!

❌ MISTAKE	✅ CORRECT
TF = $\log_{10}$(count)	**TF = 1 + $\log_{10}$(count)**
PP = P^(1/N)	**PP = (1/P)^(1/N)**
HMM = just transition	**HMM = Transition × Emission**
Viterbi: forgot emission	**Must multiply by P(word|tag) at end!**
Cosine: forgot magnitude	**Calculate BOTH ||A|| AND ||B||**
Skip-gram = context→target	**Skip-gram = target→context**

🏆 EXAM STRATEGY

Answer in this order:

1. **Q4** (TF-IDF, Cosine) - 4 marks - Direct calculation
2. **Q5** (Word2Vec) - 5 marks - Formula-based
3. **Q6** (HMM) - 4 marks - Multiplication only
4. **Q7** (Viterbi) - 5 marks - Takes time, do carefully
5. **Q2** (N-gram, PP) - 4 marks - Easy formulas
6. **Q3** (LLM) - 4 marks - Theory, relax
7. **Q1** (Intro) - 4 marks - Just write points

🍀 SHOW ALL STEPS = PARTIAL MARKS!

GOOD LUCK!

YOU'VE GOT THIS! 🎓

Remember: Show all your work for partial credit!