Natural Language Processing

Course Plan and Coursework

Ramaseshan Ramachandran

Contents

COURSE PLAN: 22 SESSIONS OF 80 MINUTES (1/2)

#	Session
1	What is NLP, why it is hard, course logistics
2	NLP tasks, corpora, preprocessing
3	Empirical laws: Zipf, Heaps, type–token ratio
4	TF-IDF, incidence matrix, similarity measures
5	Subword tokenization (BPE) + edit distance
6	Vector space models, distributional hypothesis
7	Count-based vectors (HAL/COALS) + SVD
8	Neural networks I: perceptron, activations, loss
9	Neural networks II: gradient descent, backpropagation, generalization
10	word2vec I: CBOW and skip-gram
11	word2vec II: hierarchical softmax, negative sampling; GloVe, FastText

COURSE PLAN: 22 SESSIONS OF 80 MINUTES (2/2)

#	Session
12	n-gram language models: chain rule, Markov, MLE, smoothing
13	Perplexity + neural language models
14	RNNs and backpropagation through time
15	LSTM/GRU + sequence-to-sequence
16	ELMo and the origins of attention
17	The transformer: self-attention, multi-head, positional encoding
18	BERT vs. GPT; the LLM landscape
19	Decoding strategies + prompt engineering
20	Fine-tuning and PEFT (LoRA) + alignment (RLHF)
21	Retrieval-Augmented Generation + MT recap + evaluation (BLEU $\rightarrow$ modern)
22	Bias, responsible AI, course synthesis

- **NLP Definition**
  A branch of artificial intelligence that addresses challenges in web search, question answering, language translation, and more.
- **Deep Learning Impact**
  Revolutionized NLP performance, enabling breakthroughs in understanding and generating human language.
- **Course Focus**
  Provides foundational knowledge of NLP techniques, basic machine learning techniques, modern neural network architectures, and practical experience with large corpora processing and real-world NLP task implementation.

Fundamental Techniques

Effective NLP pipelines rely on essential preprocessing steps like tokenization, sentence segmentation, stemming, lemmatization, and stop word removal, which clean and standardize text data for downstream processing and analysis.

- Tokenization and sentence boundary detection
- Stemming and lemmatization for word normalization
- Stop word removal and noise filtering

Language Models

Building probabilistic language models from preprocessed text allows statistical understanding of word sequences, frequency distributions, and linguistic patterns, paving the way for more sophisticated neural approaches to language understanding and generation.

- Frequency analysis and corpus statistics
- Text normalization and standardization techniques
- N-gram models and statistical distributions

FEATURE ENGINEERING

Classical Methods

- Bag-of-Words model for document representation
- TF-IDF weighting for term importance
- Dimensionality reduction using PCA and SVD

Word Embedding

- HAL and COALS semantic vectors
- Word2Vec (skip-gram and CBOW) and GloVe vector representations

Contextual Embedding

- Embeddings from Language Models (ELMo) contextualized word vectors
- Bidirectional Encoder Representations from Transformers (BERT)
- Dynamic embeddings for polysemous words

CLASSICAL MACHINE LEARNING

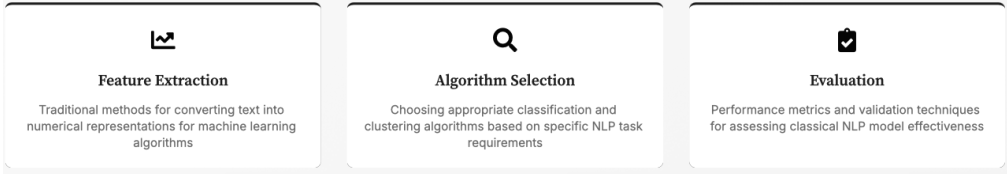
Traditional supervised and unsupervised learning approaches provide fundamental techniques for NLP classification and clustering tasks

Supervised Learning

- Classification algorithms for text categorization
- Support Vector Machines for sentiment analysis
- Naive Bayes for document classification

Unsupervised Learning

- Clustering techniques for topic discovery
- Topic modeling using LDA approaches
- Hierarchical clustering for document organization



1. **RNN Architecture**
   Recurrent Neural Networks process sequential data by maintaining hidden states that capture temporal dependencies in text sequences
2. **LSTM Networks**
   Long Short-Term Memory networks solve vanishing gradient problems through gating mechanisms enabling learning of long-range dependencies
3. **GRU Networks**
   Gated Recurrent Units provide simplified architecture with reset and update gates for efficient sequence modeling and computational performance

Attention Mechanism

- **Paper Title**: Attention Is All You Need
- **Key Innovation**: Self-attention mechanisms for parallel processing of sequences
- **Advantage**: Effectively capture long-range dependencies and overcome limitations of RNNs

Modern Applications

Transformer-based models like BERT, GPT, and T5 have achieved state-of-the-art performance across NLP benchmarks, with BERT exceeding 80% accuracy on GLUE and T5 demonstrating F1 scores above 90% on Stanford Question Answering Dataset (SQuAD) question-answering tasks.

LLM Framework

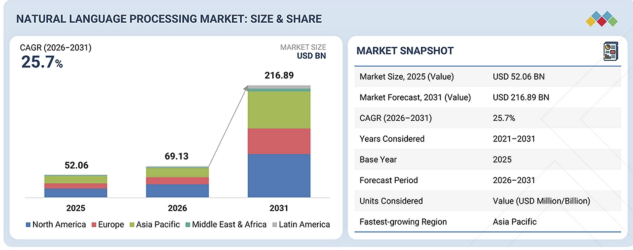
Large Language Models (LLMs) are the current state of the art in NLP. Models such as GPT-4 (rumored to have ~1.8T parameters) generate, summarize, translate, and reason over text across a wide range of domains. Newer systems (GPT-5, Claude, Gemini, DeepSeek, Llama) add long contexts, tool use, and multimodal input.

Model	Parameters
GPT-4o	1.8T (Rumored)
DeepSeek R1	671B (37B active, MoE[1])
DeepSeek V3	671B (37B active, MoE)
Grok 4	1.7T
Falcon 180B	180B

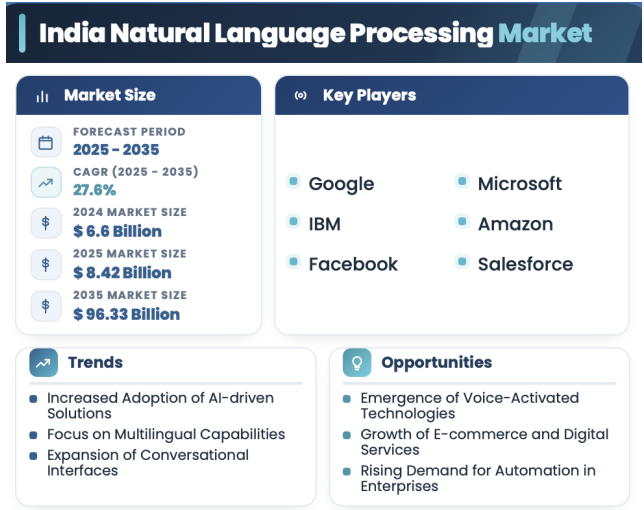
[1] Mixture of Experts

GLOBAL NLP MARKET SIZE

Large Language Models have revolutionized natural language processing, with the global NLP market projected to reach $68.1 billion by 2028. Modern LLMs like GPT-4, Claude, and Gemini excel in text understanding, generation, and reasoning, but raise ethical concerns about bias, misinformation, and responsible AI development. They demonstrate emergent abilities in few-shot learning, chain-of-thought reasoning, and multi-modal understanding.



INDIAN MARKET SIZE



Source: Indian Market Size

QUIZZES, MID AND END SEMESTER EXAMS

1. Surprise Quizzes - Marks 10
2. Mid/End Semester Exam - Marks 10
3. There will be one written examination (either MidSem or EndSem in the traditional style. This will carry maximum weightage (40 marks)

COURSE WORK: EIGHT ASSIGNMENTS - 40 MARKS

- **Eight auto-graded assignments** (100 points each), one per stage of the course:
  1. The shape of language: corpus statistics (Zipf, Heaps, TF-IDF)
  2. Build the tokenizer: BPE by hand and in code
  3. Embedding forensics: mystery vectors, analogies, bias probe
  4. Language models: n-grams, smoothing, perplexity
  5. Classification leaderboard: beat the baselines on hidden labels
  6. Transformer anatomy: attention by hand, parameters by count
  7. Steering LLMs: decoding, prompting to a target, LoRA math
  8. RAG capstone: build the retriever, ground the model
- Every assignment is designed for **8–10 hours** of real work; each of you gets a **personal variant** generated from your roll number. Copying a friend's answers is visibly wrong on *your* variant

HOW SUBMISSION WORKS

- Each assignment lives in a **private GitHub repository** created for you (`CMI-NLP/aN-<roll>`); you push your work there before the deadline
- A **validation check** runs on every push. A green tick means your submission is well-formed (it does *not* mean full marks: grading runs on hidden tests, hidden documents, and hidden labels)
- **AI tools are allowed on every assignment**. Disclose what you used in `AI_USAGE.md`. The marks live where pasted output breaks: hidden test cases, your personal numbers, and consistency checks against your own reported data
- Integrity: a short post-submission quiz about *your own* submission, and a 5-minute oral if the two disagree. Fabricated results are treated as plagiarism

---

Marks: each assignment scores /100 (90 auto-graded + 10 short written answers).

COURSE WEIGHTS