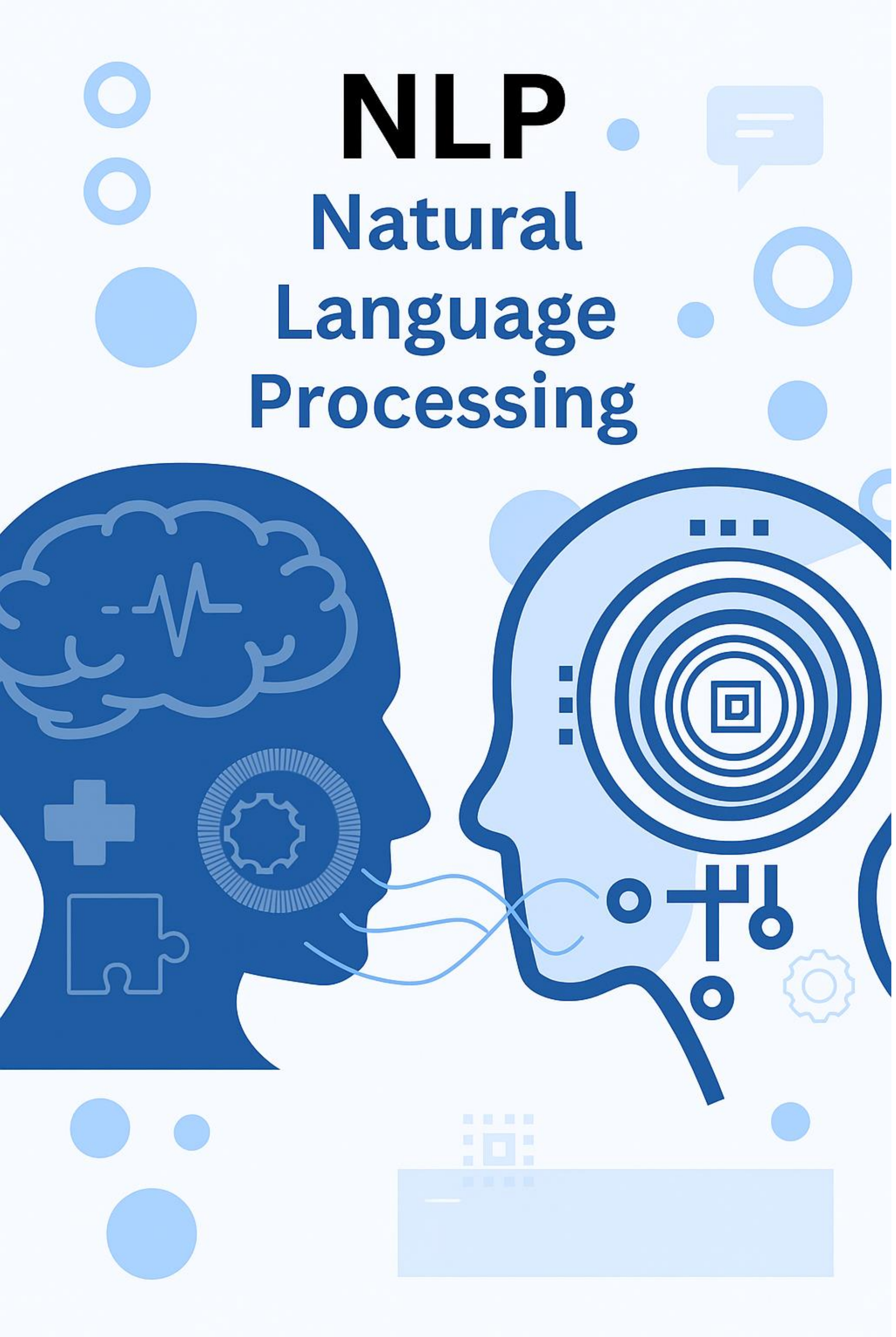

NATURAL LANGUAGE PROCESSING (NLP)

PMDS606L

MODULE 1

LECTURE 4

Dr. Kamanasish Bhattacharjee

Assistant Professor

Dept. of Analytics, SCOPE, VIT

REAL LIFE APPLICATIONS OF NLP

- Spell Checking
- Grammar Checking
- Information Extraction
- Information Retrieval
- Question Answering
- Machine Translation

SPELL CHECKING

Types of Errors

- **Non-word errors**

  Word does not exist in the dictionary

  Example: “recieve” → “receive”
- **Real-word errors**

  Word is valid but incorrect in the context

  Example: “Their going to the store” →

“They’re going to the store”

SPELL CHECKING

Components of Spell Checking

- **Error Detection**
- **Candidate Generation**
    - Edit Distance Methods
    - Phonetic Algorithms
    - Keyboard Distance
- **Candidate Ranking and Selection**

SPELL CHECKING

Input Text → Tokenization →
For each token:
→ Error Detection (dictionary lookup)
→ If misspelled:
→ Generate Candidates (edit distance, phonetics)
→ Rank Candidates (frequency, context)
→ Replace with Best Candidate

SPELL CHECKING

Applications of Spell Checking in NLP

- **Text Editors** (MS Word, Google Docs)
- **Search Engines** (“Did you mean?” suggestions)
- **Autocorrect in Smartphones**
- **Voice Assistants** (after speech-to-text conversion)
- **Chatbots and Email Filters**
- **Assistive Tools for Language Learners**

GRAMMAR CHECKING

- Modern grammar checkers go beyond rule-based correction and use **deep learning**, **contextual understanding**, and **large corpora** to detect even subtle issues.

- Grammar checkers aim to ensure syntactic and semantic correctness by analyzing sentence **structure**, **agreement rules**, and **contextual usage.**

Error Type	Example	Correction
Subject-verb agreement	He go to school	He **goes** to school
Tense errors	She eat yesterday	She **ate** yesterday
Article misuse	She has **a** idea	She has **an** idea
Preposition errors	He arrived **to** the airport	He arrived **at** the airport
Word order	Beautifully she sings	She sings beautifully
Pronoun errors	Me went there	**I** went there
Run-on sentences	I went home I slept	I went home, and I slept
Fragment sentences	Because he was late.	He was late, so...

APPROCHES IN GRAMMAR CHECKING

- Rule-Based Grammar Checking
- Statistical Grammar Checking
- Machine Learning-Based Grammar Checking
- Deep Learning-Based Grammar Correction

RULE-BASED GRAMMAR CHECKING

- Uses hand-crafted grammar rules and regular expressions.
- Simple but rigid and language-specific.
- Examples
  - **Grammarly** (initial versions)
  - **LanguageTool**
  - **Ginger** (initially rule-based)

STATISTICAL GRAMMAR CHECKING

- Use of **n-gram language models** (e.g., trigram models)
- Detect errors by measuring how likely a sentence is based on training data
- “He go to school” has a lower probability than “He goes to school”
- Examples

  Grammarly (older hybrid systems)

  Microsoft Word's statistical checker (pre-deep learning)

ML-BASED GRAMMAR CHECKING

- Supervised classification (Is this word/phrase grammatically correct?)
- Features: POS tags, dependency trees, context windows
- Examples

Google's early grammar checker

CoNLL Shared Tasks on Grammatical Error Correction (GEC)

DL-BASED GRAMMAR CHECKING

- Treat grammar correction as a **sequence-to-sequence (seq2seq)** task
- LSTM Encoder-Decoder, Transformer Encoder-Decoder, BERT-based models (e.g., RoBERTa, T5, BART, GECToR)
- Examples

  GECToR (Efficient Transformer-based model for grammar correction)

  GingerIt

  Grammarly (Advanced)

  DeepL Write

  ChatGPT, Google's grammar checker

GRAMMAR CHECKING

Applications of Grammar Checkers

- **Word processors** (Google Docs, MS Word)
- **Writing assistants** (Grammarly, Quillbot, DeepL Write)
- **Language learning apps** (Duolingo, HelloTalk)
- **Chatbots** (to ensure correct replies)
- **Speech-to-text systems** (after ASR)

GRAMMAR CHECKING

Input: “She not goes to college.”

Steps:

- **Tokenization**: [She, not, goes, to, college, .]
- **POS Tagging**: PRON, ADV, VERB, PREP, NOUN, PUNCT
- **Parse Tree**: Detected mismatch in negation and verb form
- **Error Detection**: "not goes" is ungrammatical
- **Candidate Generation**: “does not go”, “doesn’t go”, “is not going”
- **Ranking**: Based on language model → “does not go”
- **Correction**: “She does not go to college.”

INFORMATION EXTRACTION (IE)

Task	Example
Named Entity Recognition (NER)	“Barack Obama was born in Hawaii.” → Entities: **Barack Obama** (Person), **Hawaii** (Location)
Relation Extraction	“Google acquired YouTube in 2006.” → (Google, **acquired**, YouTube, 2006)
Event Extraction	“An earthquake struck Japan on Monday.” → Event: **earthquake**, Location: **Japan**, Date: **Monday**
Coreference Resolution	“Mary said she would help.” → **Mary = she**
Template Filling	“John works at Microsoft.” → Fill: {Person: John, Organization: Microsoft, Job: employee}

IE TECHNIQUES AND MODELS

Technique	Description	Example Tools
Rule-Based	Manually defined grammar and pattern-matching rules	GATE, spaCy Matcher
Statistical Models	CRF, HMM, MaxEnt	Stanford NER
Neural Models	BiLSTM-CRF, CNN, RNN	Flair, AllenNLP
Transformer-based	Pretrained language models with fine-tuning	BERT, RoBERTa, spaCy 3+, T5, GPT, REBEL

OTHER IE APPROACHES

- Distant Supervision

- Open Information Extraction (OpenIE)
  - OpenIE5
  - Stanford OpenIE
  - MinIE

- Zero-Shot IE

- Few-Shot IE

APPLICATIONS OF IE

Domain	Use Case
Search Engines	Enhancing result snippets with extracted facts
Chatbots / QA Systems	Answering fact-based queries
Healthcare	Extracting patient data from clinical notes
Finance	Extracting company earnings, merger news
Legal	Contract clause identification
Social Media Analysis	Entity and trend detection from tweets/posts

INFORMATION RETRIEVAL (IR)

- **Information Retrieval (IR)** is the process of obtaining information system resources (e.g., documents, paragraphs, sentences) that are relevant to an **information need** (query) from a large corpus.

- It is the foundation of **search engines**, **question answering systems**, and **document recommendation tools**.

INFORMATION RETRIEVAL (IR)

Task	Example
Document Retrieval	“Jurafsky” → Return documents written by Jurafsky
Passage Retrieval	“Where was Gandhi born?” → Return relevant sentence
Ad-hoc Retrieval	User poses novel, unstructured queries
Question Answering	Extract answer phrases from retrieved docs
Semantic Search	Match queries based on meaning, not just keywords

APPLICATIONS OF IR

Domain	Use Case
Search Engines	Google, Bing, DuckDuckGo
E-commerce	Product search and ranking
Chatbots	Retrieve relevant FAQs
Legal/Medical NLP	Find related case laws or symptoms
Document Recommendation	Suggest papers, news, or books
Question Answering (QA)	Retrieve evidence for answering queries

IR MODELS

Model	Description
BERT (re-ranker)	Contextual ranking of documents after initial retrieval (not suitable for large-scale retrieval)
Siamese BERT / Sentence-BERT (SBERT)	Converts queries and documents into fixed-length semantic vectors for fast search
Dense Passage Retrieval (DPR)	Dual BERT encoders for questions and passages; used in QA
ColBERT (Contextual Late Interaction)	Efficient dense retrieval using token-level interactions
TAS-B (Token-Aware SBERT)	Hybrid model combining BERT and interaction layers for better performance
ANCE (Approximate Nearest Neighbor Negative Contrastive Estimation)	Learns better dense representations for fast retrieval
RAG (Retrieval-Augmented Generation)	Combines dense retrieval and text generation (e.g., BART + FAISS)

QUESTION ANSWERING (QA)

Type	Description	Example
Closed-domain QA	Focused on a specific subject area (e.g., medicine, law)	“What is the normal blood pressure?”
Open-domain QA	Answers any general question from a large corpus like Wikipedia	“Who discovered penicillin?”
Factoid QA	Provides a factual answer (name, date, location)	“When did World War II end?” → “1945”
Yes/No QA	Returns binary responses	“Is Mount Everest the tallest mountain?”
List QA	Returns a list of answers	“Name the continents”
Generative QA	Produces natural language answers (beyond span extraction)	“Why is the sky blue?” → "Because molecules in the air scatter blue light more."

QUESTION ANSWERING (QA) PIPELINE

User Question

↓

[1] Information Retrieval (IR)

↓

Top-k Relevant Passages or Documents

↓

[2] Machine Reading Comprehension (MRC)

↓

Extracted or Generated Answer

QUESTION ANSWERING (QA) MODELS

Model	Use Case	Type
BERT	Extractive QA from short passages	Span-based
RoBERTa / XLNet	Improved versions of BERT	Span-based
T5 (Text-to-Text Transfer Transformer)	Generative QA	Seq2Seq
RAG (Retrieval-Augmented Generation)	Combines retrieval + generation	Hybrid
GPT-3 / GPT-4 / Claude / LLaMA	Zero-shot QA, open-domain	Generative
DPR (Dense Passage Retrieval)	Open-domain retrieval	Dual Encoder

QA SYSTEMS

System	Platform
Google Search Snippets	Open-domain QA
Alexa / Siri / Google Assistant	Voice-based QA
IBM Watson	Domain-specific QA
ChatGPT / Claude / Bard	Large Language Model-based QA
Haystack QA Pipelines	End-to-end Python QA systems

MACHINE TRANSLATION (MT) PIPELINE

Source Language Text
↓
1. Text Preprocessing
↓
2. Tokenization & Subword Segmentation
↓
3. Embedding Layer (Encoder Input)
↓
4. Encoder (e.g., Transformer)
↓
5. Attention Mechanism
↓
6. Decoder (Auto-regressive)
↓
7. Target Language Generation
↓
8. Postprocessing (Detokenization, Grammar Fixes)
↓
Target Language Text

DL-BASED MT

Feature	Description
Uses	Neural networks (especially Transformers)
Models	Encoder–Decoder architecture
Handles	Context, long dependencies, fluency
Examples	Google Translate (Transformer), DeepL (custom NMT), OpenNMT
Frameworks	OpenNMT, Fairseq, MarianNMT, HuggingFace Transformers

APPLICATIONS OF MACHINE TRANSLATION

Domain	Use
Web Translation	Google Translate, DeepL
E-commerce	Translate reviews, product listings
Healthcare	Cross-lingual patient communication
Legal / Policy	Document translation between jurisdictions
News & Media	Multilingual publishing
Government	Translation of official documents
Social Media	Facebook auto-translation of posts

TOOLS AND LIBRARIES FOR MT

Tool	Description
OpenNMT	Open-source NMT toolkit (PyTorch, TensorFlow)
MarianNMT	Fast NMT framework used by Microsoft
Fairseq	Facebook's NLP toolkit for sequence modeling
Transformers (HuggingFace)	Pretrained MT models like T5, mBART, MarianMT
Google Translate API	Commercial translation service
DeepL API	High-quality translations in European languages
Bergamot / Firefox Translator	Offline NMT in browser (privacy-preserving)