Unit-IV

Natural Language Generation

Natural Language Generation (NLG) simply means producing text from computer data.

It acts as a translator and converts the computerized data into natural language representation.

In this, a conclusion or text is generated based on collected data and input provided by the user.

It is the natural language processing task of generating natural language from a machine representation system. Natural Language Generation in a way acts contrary to Natural language understanding.

In natural language understanding the system needs to disambiguate the input sentence to produce the machine representation language, whereas in Natural Language Generation the system needs to make decisions about how to put a concept into words.

Ex:

NLG system is the Pollen Forecast for Scotland system which could essentially be a template. NLG system takes as input six numbers, which predict the pollen levels in different parts of Scotland. From these numbers, a short textual summary of pollen levels is generated by the system as its output.

How NLG works?

Natural Language Generation (NLG) is a branch of AI that focuses on the automatic generation of human-like language from data. NLG systems take structured data as input and convert it into coherent, contextually relevant human-readable text. The goal is for the generated text to sound like it was written by a human.

Here’s a high-level overview of how Natural Language Generation works:

Data Input: Structured data is the first input used by NLG systems. This information may originate from a number of sources, including spreadsheets, databases, and other organized formats.

Content Planning: Based on an analysis of the input data, the system decides what details to include in the text that is generated. Making choices regarding the selection of content, arrangement, and general structure is required.

Text Planning: The NLG system arranges the content's natural language expression after it has been decided upon. It chooses the right wording, tone, and style for the text that is generated.

Sentence Generation: Using the planned content as a guide, the system generates individual sentences. Choosing the right words, phrases, and syntactic structures is necessary for this. While some NLG systems generate text using pre-defined templates, others might use more advanced techniques like machine learning.

Coherence and Consistency: Text produced by NLG systems should be consistent and coherent. This entails making certain that the sentences that are produced follow grammatical and stylistic conventions and flow naturally. It might also entail continuing to produce content that is consistent with earlier works.

Refinement: To raise the calibre of the produced text, a refinement procedure may be used. This could entail doing extra proofreading for naturalness, clarity, and grammar. -------------------------

Architectures of NLG Systems

Traditionally this has been approached by discretely implementing the three stages and assembling them into a pipeline of some sort.

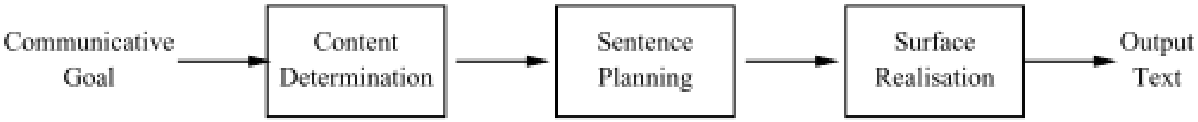

Figure 3.1: The three Reiter processes in a pipeline architecture

- **Content determination:** Deciding the main content to be represented in a sentence or the information to mention in the text.

  Ex: Deciding whether to explicitly mention that pollen level is 7 in the south-east.

- **Document structuring:** Deciding the structure or organization of the conveyed information.

  For example, deciding to describe the areas with high pollen levels first, instead of the areas with low pollen levels.

- **Sentence Planning:** Putting of similar sentences together to improve understanding and readability.

  For instance, merging the two sentences Grass pollen levels for Friday have increased from the moderate to high levels of yesterday and Grass pollen levels will be around 6 to 7 across most parts of the country

  It is represented in single sentence form as Grass pollen levels for Friday have increased from the moderate to high levels of yesterday with values of around 6 to 7 across most parts of the country.

- **Realisation:** Creating and optimizing the text that should be correct as per the rules of grammar. For example, using will be for the future tense of to be.

A number of different architectures have emerged to combat these problems:

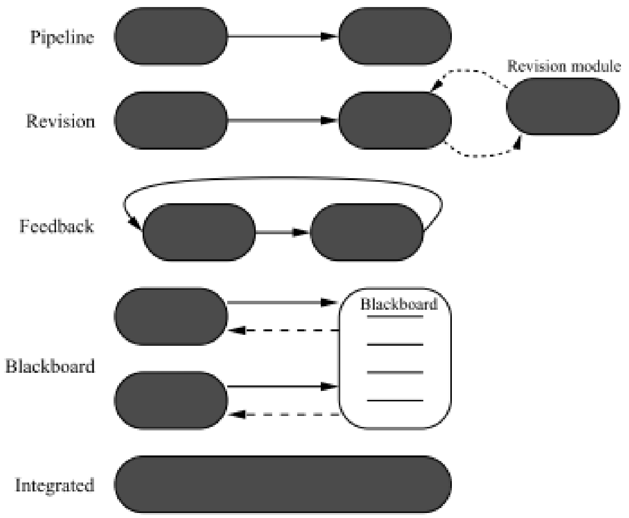

Figure 3.2: Variety of NLG system architectures

- The **revision approach** combat problems by iteratively fixing each stage in hopes of finding something better.
- **Feedback** simply feeds findings from later stages back into earlier stages, which can open up options that would otherwise be pruned away.
- **blackboard** is particularly interesting because it uses a common space where results from different stages are posted so every stage has access to what's already out there.
- But without a clear communicative goal or end user in mind, the **integrated** approach looks most promising.

Techniques for Evaluating NLG systems

1. **Task-based evaluation:** It includes human-based evaluation, which assesses how well it helps him perform a task. For example, a system which generates summaries of medical data can be evaluated by giving these summaries to doctors and assessing whether the summaries help doctors make better decisions.

2. **Human ratings:** It assess the generated text on the basis of ratings given by a person on the quality and usefulness of the text.

3. **Metrics:** It compares generated texts to texts written by professionals.

Applications of Natural Language Generation (NLG)

Natural language generation, or NLG, has numerous important uses in a range of sectors. The following are a few notable domains where NLG is extensively employed

Intelligent Automation and Reporting:

- NLG is used to transform analytical and complex data into reports and summaries that are understandable to humans. This makes it especially easy for stakeholders to comprehend and act upon insights in business intelligence.

Marketing and Content Creation:

- NLG is used to create content for blogs, websites, and advertising collateral. It can produce written materials at scale, including product descriptions and promotional content.

Virtual assistants and chatbots:

- By allowing chatbots and virtual assistants to respond in natural language, natural language generation (NLG) improves their conversational skills. Ensuring a user experience that is both engaging and human-like is imperative.

Analysis of Finance and Investments:

- Using numerical data and trends, natural language generation (NGL) is used in the finance industry to automatically produce financial reports, investment summaries, and market commentary.

Medical Records:

- NLG is used to generate medical reports, documentation, and patient summaries from electronic health records (EHR). In medical settings, it can simplify the documentation procedure.

Educational Content and E-Learning:

- NLG contributes to the creation of instructional materials, assessments, and personalized feedback for students. It aids in the creation of learning platforms that are adaptable.

Machine Translation:

Machine translation is a computer program which is design to translate text from one language (source language) to another language (target language) with- out the help of human.

Machine translation in natural language processing (NLP) refers to the automated process of translating text or speech from one language to another using computational method. This technology is essential for breaking down language barriers and enabling communication across different linguistic communities.

There are several approaches to machine translation, including:

1. Rule-based machine translation (RBMT): This approach relies on linguistic rules and dictionaries to translate text. It involves analyzing the input text, breaking it down into linguistic components, and applying grammar and syntax rules to generate the translated output. While rule-based systems can produce accurate translations for certain languages and domains, they often struggle with complex linguistic structures and idiomatic expressions.

2. Statistical machine translation (SMT): SMT uses statistical models trained on large corpora of parallel texts in different languages. These models learn to identify patterns and correlations between words and phrases in the source and target languages, allowing them to generate translations based on probabilistic principles. While SMT systems can produce reasonably accurate translations, they may struggle with low-resource languages and often require substantial amounts of training data.

3. Neural machine translation (NMT): NMT represents a paradigm shift in machine translation, employing deep learning techniques to directly model the mapping between source and target languages. NMT systems consist of

neural networks, such as recurrent neural networks (RNNs) or transformer models, that learn to encode the input text into a continuous representation and decode it into the target language. NMT has shown significant improvements in translation quality over previous approaches, especially for languages with complex syntax and semantics.

4. Transfer learning and multilingual models: Recent advancements in NLP have led to the development of transfer learning techniques and multilingual models that can translate between multiple language pairs using a single unified architecture. These models leverage large-scale pre training on diverse linguistic data and fine-tuning on specific translation tasks, allowing them to achieve state-of-the-art performance across a wide range of languages.

Problems in Machine Translation:

Machine translation in natural language processing (NLP) faces several challenges, which can affect the accuracy, fluency, and overall quality of translations. Some of the prominent problems in machine translation include:

1. **Linguistic Complexity**: Languages exhibit diverse syntactic, morphological, and semantic structures, making translation challenging. For instance, word order, grammatical rules, and idiomatic expressions vary across languages, posing difficulties for machine translation systems to accurately capture and translate these linguistic phenomena.

2. **Ambiguity**: Natural languages often contain ambiguous words, phrases, and sentences that can have multiple meanings depending on context. Resolving ambiguity is crucial for producing accurate translations, but it remains a challenging task for machine translation systems, particularly in cases where context is sparse or ambiguous.

3. **Out-of-Vocabulary (OOV) Words**: Machine translation systems may encounter words that are not present in their vocabulary or training data, known as out-of-vocabulary words. Handling OOV words requires robust

techniques for word alignment, vocabulary expansion, and handling unknown tokens to ensure accurate translations.

4. **Domain Adaptation**: Machine translation systems trained on generic corpora may struggle to translate texts in specialized domains, such as technical, medical, or legal documents, due to domain-specific terminology, jargon, and discourse patterns. Domain adaptation techniques are necessary to fine-tune translation models on domain-specific data and improve their performance in specialized domains.

5. **Data Sparsity**: Adequate training data is essential for building accurate machine translation models, but many language pairs suffer from data scarcity, especially for low-resource languages. Data augmentation, transfer learning, and semi-supervised learning techniques can help mitigate data sparsity issues and improve translation quality for under-resourced languages.

6. **Syntactic and Semantic Divergence**: Languages may exhibit syntactic and semantic differences that pose challenges for alignment and translation. For instance, languages may have different word order, syntactic constructions, and semantic representations, requiring translation models to effectively capture and preserve these differences to produce fluent and accurate translations.

7. **Cultural and Contextual Nuances**: Cultural and contextual factors influence language use and meaning, leading to challenges in translation, particularly for idiomatic expressions, humor, and cultural references. Machine translation systems must account for cultural and contextual nuances to produce culturally sensitive and contextually appropriate translations.

8. **Evaluation Metrics**: Assessing the quality of machine translations poses challenges due to the subjective nature of language and translation. Common evaluation metrics, such as BLEU (Bilingual Evaluation Understudy) and METEOR (Metric for Evaluation of Translation with Explicit Ordering), may not always correlate well with human judgments, leading to limitations in evaluating translation quality accurately.

Characteristics of Indian Languages:

Indian languages pose unique challenges and exhibit specific characteristics that impact natural language processing (NLP) tasks. Some of the key characteristics of Indian languages in the context of NLP include:

1. **Morphological Complexity**: Indian languages often exhibit rich morphological structures with complex inflectional and derivational patterns. This complexity arises from features such as agglutination, compounding, and sandhi (phonological changes at morpheme boundaries), which pose challenges for tasks like morphological analysis, stemming, and part-of-speech tagging.

2. **Script Diversity**: Indian languages are written in diverse scripts, including Devanagari, Tamil, Telugu, Bengali, Gujarati, and others. Each script has its own set of characters and orthographic conventions, requiring specialized tokenization, text normalization, and script conversion techniques to handle text input in different scripts.

3. **Lack of Standardization**: Indian languages often lack standardized spelling, grammar, and vocabulary, leading to variations in language usage across regions and dialects. This variation presents challenges for tasks such as named entity recognition, sentiment analysis, and machine translation, which rely on consistent language patterns and structures.

4. **Code-Switching and Multilinguality**: Multilingualism and code-switching are common phenomena in India, where speakers frequently mix multiple languages within a single utterance or document. Handling code-switching presents challenges for tasks like language identification, named entity recognition, and sentiment analysis, which require robust models capable of processing mixed-language input and identifying language boundaries.

5. **Low-Resource Languages**: Many Indian languages are considered low-resource in the context of NLP, with limited annotated data, resources, and tools available for language processing tasks. Addressing the needs of low-resource languages requires techniques such as transfer learning,

unsupervised learning, and data augmentation to build effective NLP models with limited training data.

6. **Semantic Richness**: Indian languages often exhibit semantic richness and expressivity, with complex lexical semantics, compound words, and idiomatic expressions. Capturing and representing this semantic complexity is crucial for tasks like semantic parsing, sentiment analysis, and question answering in Indian languages.
7. **Orthographic Variation**: Indian languages may exhibit orthographic variations due to historical, phonological, and dialectal differences. Variations in spelling, pronunciation, and word forms pose challenges for tasks like named entity recognition, word sense disambiguation, and information retrieval, which require robust models capable of handling orthographic variation.
8. **Cultural and Sociolinguistic Context**: Indian languages are embedded within diverse cultural and sociolinguistic contexts, reflecting the country's rich linguistic and cultural heritage. Understanding and incorporating cultural and sociolinguistic factors are essential for building culturally sensitive and contextually appropriate NLP applications for Indian languages.

Machine Translation Approaches:

Machine translation in natural language processing (NLP) employs various approaches, each with its own techniques and methodologies. Some of the key machine translation approaches include:

1. **Rule-Based Machine Translation (RBMT)**:
    - **Approach**: RBMT relies on linguistic rules and dictionaries to translate text from a source language to a target language. These rules capture syntactic, morphological, and semantic patterns of the languages involved.

- **Techniques**: RBMT systems typically involve morphological analysis, syntactic parsing, transfer rules, and morphological generation.
- **Advantages**: RBMT can produce accurate translations, especially for languages with clear grammar rules and well-defined linguistic structures.
- **Limitations**: RBMT systems may struggle with handling ambiguities, idiomatic expressions, and languages with complex morphology.

2. **Statistical Machine Translation (SMT)**:
    - **Approach**: SMT uses statistical models trained on large bilingual corpora to learn the mapping between source and target language phrases or sentences.
    - **Techniques**: SMT involves phrase-based models, word alignment algorithms (e.g., IBM models), language models, and decoding algorithms (e.g., beam search).
    - **Advantages**: SMT can handle a wide range of language pairs and is effective for translating colloquial language. It also benefits from data-driven learning.
    - **Limitations**: SMT may struggle with handling long-range dependencies, preserving context, and producing fluent translations, especially for languages with different word orders.
3. **Neural Machine Translation (NMT)**:
    - **Approach**: NMT represents a paradigm shift in machine translation, employing deep learning techniques to directly model the mapping between source and target languages.
    - **Techniques**: NMT systems use neural network architectures such as sequence-to-sequence models, attention mechanisms, and transformer models.

- **Advantages**: NMT has shown significant improvements in translation quality, fluency, and handling of long-range dependencies. It can effectively capture context and produce more natural-sounding translations.
- **Limitations**: NMT models require large amounts of training data and computational resources. They may also struggle with translating rare or out-of-vocabulary words.

4. **Hybrid Approaches**:
    - **Approach**: Hybrid machine translation systems combine multiple techniques, such as rule-based, statistical, and neural approaches, to leverage the strengths of each.
    - **Techniques**: Hybrid systems may incorporate rule-based pre- and post-processing, statistical models for phrase alignment, and neural networks for re-ranking or fine-tuning.
    - **Advantages**: Hybrid approaches can achieve better translation accuracy and fluency by combining complementary techniques.
    - **Limitations**: Hybrid systems can be complex to develop and maintain, requiring integration of disparate components and careful tuning of parameters.

5. **Transfer Learning and Multilingual Models**:
    - **Approach**: Transfer learning techniques and multilingual models leverage pretraining on large-scale multilingual corpora to improve translation quality across multiple language pairs.
    - **Techniques**: Transfer learning involves pretraining on a large source domain and fine-tuning on a smaller target domain or language pair. Multilingual models jointly learn representations for multiple languages and tasks.

- **Advantages**: Transfer learning and multilingual models can improve translation quality, especially for low-resource languages, by leveraging shared linguistic knowledge across languages.
- **Limitations**: Transfer learning requires sufficient overlap between the source and target domains or languages. Multilingual models may face challenges in handling language-specific nuances and idiosyncrasies.

Direct Machine Translation:

Direct machine translation in natural language processing (NLP) refers to the process of translating text directly from one language to another without relying on intermediate representations or alignments. Unlike traditional approaches such as rule-based or statistical machine translation, direct machine translation models directly learn the mapping between source and target languages using neural network architectures.

Here are some key characteristics and aspects of direct machine translation in NLP:

1. **Neural Network Architectures**: Direct machine translation models are typically based on neural network architectures, such as sequence-to-sequence models or transformer models. These models encode the source text into a fixed-length vector representation (encoder) and decode it into the target language (decoder) using attention mechanisms to focus on relevant parts of the input during decoding.
2. **End-to-End Translation**: Direct machine translation models enable end-to-end translation without relying on explicit linguistic rules or alignments. They learn to automatically align and translate source and target language sequences, making them more flexible and capable of capturing complex linguistic patterns.
3. **Attention Mechanisms**: Attention mechanisms play a crucial role in direct machine translation models by allowing the decoder to focus on relevant

parts of the source text during translation. This enables the model to capture long-range dependencies and improve translation quality, especially for languages with different word orders or complex syntactic structures.

4. **Training Data**: Direct machine translation models require large parallel corpora of translated text pairs for training. These corpora serve as the basis for learning the mapping between source and target languages and optimizing model parameters to minimize translation errors.

5. **Multilingual Models**: Direct machine translation models can be trained to translate between multiple language pairs simultaneously using a single unified architecture. Multilingual models leverage shared representations across languages to improve translation quality and efficiency, especially for low-resource languages.

6. **Fine-Tuning and Transfer Learning**: Direct machine translation models can be fine-tuned on specific language pairs or domains to further improve translation quality. Transfer learning techniques allow pre-trained models to be adapted to new languages or domains with limited annotated data, leveraging knowledge learned from other tasks or languages.

7. **Evaluation Metrics**: Direct machine translation models are typically evaluated using standard metrics such as BLEU (Bilingual Evaluation Understudy) or METEOR (Metric for Evaluation of Translation with Explicit Ordering). These metrics assess the quality of translations by comparing them to reference translations provided by human annotators.

Corpus based Machine Translation:

Corpus-based machine translation (CBMT) is a paradigm in natural language processing (NLP) that relies on large parallel corpora of translated texts to generate translations from one language to another. Unlike rule-based approaches that depend on handcrafted linguistic rules or statistical methods that analyze frequency distributions, CBMT systems directly use the knowledge embedded in the parallel corpora to produce translations. Here's an overview of corpus-based machine translation in NLP:

1. **Parallel Corpora**: CBMT systems require access to substantial parallel corpora containing aligned text in the source and target languages. These corpora serve as the primary source of training data, allowing the system to learn the correspondence between phrases or sentences in different languages.

2. **Alignment**: The first step in CBMT involves aligning corresponding segments of text in the parallel corpora. This alignment process identifies which parts of the source text correspond to which parts of the target text, enabling the system to learn translation patterns and relationships.

3. **Phrase-Based Translation**: CBMT systems often operate at the level of phrases or short segments of text rather than individual words. During translation, the system identifies matching or similar phrases in the source and target languages and selects the translation that best fits the context.

4. **Statistical Models**: While CBMT is based on parallel corpora, statistical models may still play a role in the translation process. For example, the system may use statistical techniques to estimate translation probabilities or to rank alternative translations based on their likelihood given the training data.

5. **Lexical and Structural Correspondence**: CBMT systems aim to capture both lexical and structural correspondences between languages. Lexical correspondences involve mapping words or phrases from the source language to their counterparts in the target language, while structural correspondences involve preserving the syntactic and semantic relationships between elements of the translated text.

6. **Limitations**: CBMT systems may face limitations in handling ambiguity, idiomatic expressions, and domain-specific language use. Since they rely heavily on the patterns found in the training data, they may struggle with translating text outside the scope of their training corpus or with producing fluent and natural-sounding translations in certain contexts.

7. **Evaluation**: CBMT systems are evaluated using standard metrics such as BLEU (Bilingual Evaluation Understudy) or METEOR (Metric for

Evaluation of Translation with Explicit Ordering). These metrics compare the output of the system against reference translations provided by human translators to assess translation quality.

Semantic or knowledge-based machine translation (KBMT):

Semantic or knowledge-based machine translation (KBMT) systems in natural language processing (NLP) leverage semantic representations or external knowledge sources to enhance the translation process. Unlike traditional statistical or neural machine translation (NMT) approaches that primarily rely on large parallel corpora, KBMT systems incorporate semantic knowledge about the meaning and structure of languages to improve translation quality. Here's an overview of semantic or knowledge-based MT systems in NLP:

1. **Semantic Representations**: KBMT systems utilize semantic representations of text to capture the meaning and context of sentences in both the source and target languages. These representations can be obtained through various techniques, including semantic parsing, semantic role labeling, or distributed word embeddings (e.g., word2vec, GloVe).

2. **Semantic Analysis**: Before translating a sentence, KBMT systems perform semantic analysis to extract and represent the underlying meaning of the input text. This may involve identifying entities, relationships, events, and other semantic elements present in the text.

3. **Knowledge Integration**: KBMT systems integrate external knowledge sources, such as ontologies, lexicons, or knowledge graphs, into the translation process. These knowledge sources provide additional semantic context and constraints that guide the translation process and help disambiguate ambiguous phrases or terms.

4. **Semantic Matching**: During translation, KBMT systems use semantic matching techniques to align semantically similar phrases or concepts between the source and target languages. This allows the system to produce translations that preserve the intended meaning and convey the semantic nuances of the original text.

5. **Domain-Specific Knowledge**: KBMT systems can leverage domain-specific knowledge or specialized terminology to improve translation accuracy in specific domains or industries. By incorporating domain-specific lexicons, ontologies, or terminology databases, the system can produce more accurate and contextually appropriate translations for specialized texts.

6. **Interlingual Representations**: Some KBMT systems use interlingual representations, which represent the meaning of a sentence in a language-independent form. By translating both the source and target sentences into a common interlingua, the system can perform translation at the semantic level, facilitating accurate and fluent translations across different language pairs.

7. **Evaluation Metrics**: KBMT systems are evaluated using standard metrics such as BLEU (Bilingual Evaluation Understudy) or METEOR (Metric for Evaluation of Translation with Explicit Ordering). However, since KBMT systems focus on capturing semantic accuracy and preserving meaning, additional evaluation criteria related to semantic fidelity, fluency, and adequacy may also be used.

--

Translation involving Indian Languages:

Translation involving Indian languages in natural language processing (NLP) presents several unique challenges and opportunities due to the linguistic diversity and complexity of Indian languages. Here are some key aspects of translation involving Indian languages in NLP:

1. **Script Diversity**: Indian languages are written in various scripts, including Devanagari, Tamil, Telugu, Bengali, Gujarati, and others. Translation systems need to be capable of processing text in different scripts and handling script conversion when translating between languages with different writing systems.

2. **Morphological Complexity**: Indian languages often exhibit rich morphological structures with complex inflectional and derivational patterns. Translation systems must handle the morphological complexity of

Indian languages, including word inflections, verb conjugations, and compound words.

3. **Lexical Variation**: Indian languages may have significant lexical variation across regions and dialects. Translation systems need to account for regional variations in vocabulary, idiomatic expressions, and colloquialisms to produce accurate and contextually appropriate translations.

4. **Code-Switching and Multilingualism**: Code-switching and multilingualism are common phenomena in India, where speakers frequently mix multiple languages within a single utterance or document. Translation systems must be capable of handling code-switching and translating mixed-language input accurately.

5. **Low-Resource Languages**: Many Indian languages are considered low-resource in the context of NLP, with limited annotated data and linguistic resources available. Translation systems need to address the challenges of data scarcity and develop techniques for effectively translating low-resource languages.

6. **Cultural and Sociolinguistic Context**: Indian languages are embedded within diverse cultural and sociolinguistic contexts, reflecting the country's rich linguistic and cultural heritage. Translation systems must consider cultural nuances, social conventions, and contextual factors to produce culturally sensitive and contextually appropriate translations.

7. **Machine Translation Approaches**: Various machine translation approaches, including statistical machine translation (SMT), neural machine translation (NMT), and hybrid approaches, are used for translating Indian languages in NLP. NMT models, in particular, have shown promising results in improving translation quality and fluency for Indian languages.

8. **Resource Development**: Developing comprehensive linguistic resources, such as parallel corpora, bilingual lexicons, and annotated datasets, is essential for training and evaluating machine translation systems for Indian languages. Efforts are underway to create and curate linguistic resources for Indian languages to support research and development in NLP.