PBR VITS (AUTONOMOUS)

III B.TECH CSE-AI

NATURAL LANGUAGE PROCESSING

TEXT BOOK: James Allen, **Natural Language Understanding**, 2nd Edition, 2003, Pearson Education

Course Instructor: Dr KV Subbaiah, M.Tech, Ph.D, Professor, Dept. of CSE

UNIT–V Machine Translation and Multilingual Information

Machine Translation Survey: Introduction, Problems of Machine Translation, Is Machine Translation Possible, Brief History, Possible Approaches, Current Status. Anusaraka or Language Accessor: Background, Cutting the Gordian Knot, The Problem, Structure of Anusaraka System, User Interface, Linguistic Area, Giving up Agreement in Anusaraka Output, Language Bridges.
Multilingual Information Retrieval - Introduction, Document Pre-processing, Monolingual Information Retrieval, CLIR, MLIR, Evaluation in Information Retrieval, Tools, Software and Resources.
Multilingual Automatic Summarization - Introduction, Approaches to Summarization, Evaluation, How to Build a Summarizer, Competitions and Datasets.

---

1. Introduction:

Natural Language Processing (NLP) has revolutionized the field of machine translation, enabling the development of advanced algorithms and models that can automatically translate text or speech from one language to another. This survey focuses on gathering insights and opinions specifically related to machine translation in the context of NLP.

Machine translation plays a vital role in bridging language barriers and promoting global communication. It has applications in various domains, including multilingual customer support, cross-border business collaborations, and information dissemination on the internet. As NLP techniques continue to advance, machine translation systems have become more accurate and efficient, improving the overall quality of translations.

This survey aims to assess the current state of machine translation in the context of NLP, explore the challenges faced by researchers and practitioners,

and identify potential areas for future research and development. By participating in this survey, you will contribute to the understanding of the current landscape of machine translation in the field of NLP.

Your responses will be treated anonymously, and the survey results will be used for research purposes only. Please provide your insights and opinions based on your knowledge and expertise in the field of NLP and machine translation.

2. Problems of Machine Translation

Machine translation in NLP faces several challenges that impact the quality and accuracy of translations. Some of the prominent problems include:

1. **Ambiguity:** Natural languages often contain ambiguous words, phrases, or sentences that can have multiple interpretations. Machine translation systems may struggle to accurately disambiguate such instances, leading to incorrect translations.
2. **Idiomatic Expressions and Cultural Nuances**: Languages contain idiomatic expressions and cultural nuances that are challenging to translate accurately. Machine translation systems may struggle to capture the intended meaning or may produce literal translations that lack the cultural context, resulting in translations that sound awkward or are incorrect.
3. **Syntax and Grammar:** Translating sentences while preserving the correct syntax and grammar is a complex task. Machine translation systems may produce translations that have grammatical errors, incorrect word order, or lack fluency, making them harder to understand.
4. **Out-of-vocabulary (OOV) Words:** Machine translation systems often encounter words or phrases that are not present in their training data. These out-of-vocabulary words pose a challenge as the system may not have learned their translations, leading to inaccurate or untranslated words in the output.
5. **Domain-specific Terminology:** Different domains have their own specific terminologies, such as technical or medical terms, which might not have direct translations. Machine translation systems may struggle to handle these specialized terms, resulting in inaccurate or inconsistent translations.
6. **Language Pair Discrepancies:** The performance of machine translation systems can vary significantly depending on the language pair being translated. Some language pairs have more training data available, leading to better performance, while others may have limited resources, resulting in poorer translations.
7. **Lack of Context Understanding**: Machine translation systems often lack the ability to understand the broader context of a sentence or document. They may struggle with pronoun resolution, coreference resolution, or understanding context-dependent meanings, leading to translation errors.

8. **Rare or Low-Resource Languages:** Machine translation for rare or low-resource languages poses additional challenges due to limited training data and resources. It is often difficult to achieve high-quality translations for these languages due to the scarcity of linguistic resources and models trained specifically for them.

Addressing these challenges requires ongoing research and development in the field of machine translation in NLP. By tackling these problems, we can improve the accuracy, fluency, and overall quality of machine translation systems.

3. Is Machine Translation Possible

Yes, machine translation is a subfield of Natural Language Processing (NLP). Machine translation involves the use of algorithms and computational methods to automatically translate text or speech from one language to another. It aims to bridge the language barrier and facilitate communication between people who speak different languages.

Machine translation can be achieved through various approaches, including rule-based systems, statistical models, and more recently, neural machine translation (NMT). NMT, powered by deep learning techniques, has shown significant improvements in translation quality compared to earlier approaches.

NLP encompasses a wide range of tasks beyond machine translation, such as language understanding, sentiment analysis, question answering, text summarization, and many others. However, machine translation remains one of the fundamental and widely studied applications within the field of NLP.

4. Current Status

As of my knowledge cutoff in September 2021, machine translation has made significant progress in the field of NLP, particularly with the advent of neural machine translation (NMT) models. NMT has revolutionized the quality of machine translation outputs and has become the dominant approach in recent years.

NMT models employ deep learning techniques, specifically neural networks, to learn the mapping between different languages. These models can handle complex sentence structures, capture context dependencies, and generate more fluent translations compared to previous statistical and rule-based methods.

The most notable NMT architecture is the Transformer model, introduced in 2017. Transformers leverage attention mechanisms to focus on relevant parts of the input sequence during translation, allowing for better long-range dependencies modeling. This architecture has achieved state-of-the-art results on various machine translation benchmarks.

Additionally, large-scale pretraining techniques, such as unsupervised or semi-supervised learning, have been explored to enhance machine translation performance. By leveraging massive amounts of monolingual data, these approaches can improve translation quality even without parallel corpora.

It's important to note that the field of NLP is rapidly evolving, and new techniques and models may have been developed since my last update. I recommend referring to recent research papers, conferences, and advancements in the field to stay up to date with the current state of machine translation in NLP.

5. Cutting the Gordian Knot

"Cutting the Gordian Knot" is a metaphorical expression that originates from the legend of Alexander the Great. According to the legend, Alexander encountered a complex knot tied by Gordius, the king of Phrygia. The knot was said to be impossible to untie, and it was prophesied that whoever could unravel it would become the ruler of Asia. Instead of attempting to untie the knot, Alexander famously took his sword and sliced through it, "cutting the Gordian Knot" and solving the problem in a bold and unconventional way.

In the context of NLP, "cutting the Gordian Knot" can refer to finding a simple and effective solution to a complex problem or challenge. NLP tasks often involve intricate linguistic nuances, ambiguity, and challenges related to language understanding and generation. Researchers and practitioners in NLP continuously seek innovative approaches and techniques to tackle these challenges.

The expression "cutting the Gordian Knot" in NLP may signify the discovery of a breakthrough technique, a novel algorithm, or an innovative model architecture that simplifies or solves a previously difficult or unsolved problem in natural language processing.

It's worth noting that specific techniques and approaches for "cutting the Gordian Knot" in NLP may vary depending on the particular problem or task at hand. Researchers and practitioners employ a range of methods, including deep learning, neural networks, transfer learning, reinforcement learning, and more, to overcome challenges and improve the performance of NLP systems.

---

6. Structure of Anusaraka System

The name Anusaaraka is derived from Sanskrit word Anusaaran that means “to follow”. In the processing of Anusaaraka output appears in one step followed by the next one. Hence it is named so based on its way of generating the output. Anusaaraka is a translator that accepts English as input and produces output in Telugu/Hindi etc. The sentence is passed through various stages of defragmentation and analysis before the output is generated.

The Anusaaraka architecture has been designed and developed based on issues revealed during an evaluation of conventional machine translation. The architecture is shown in Fig 1.

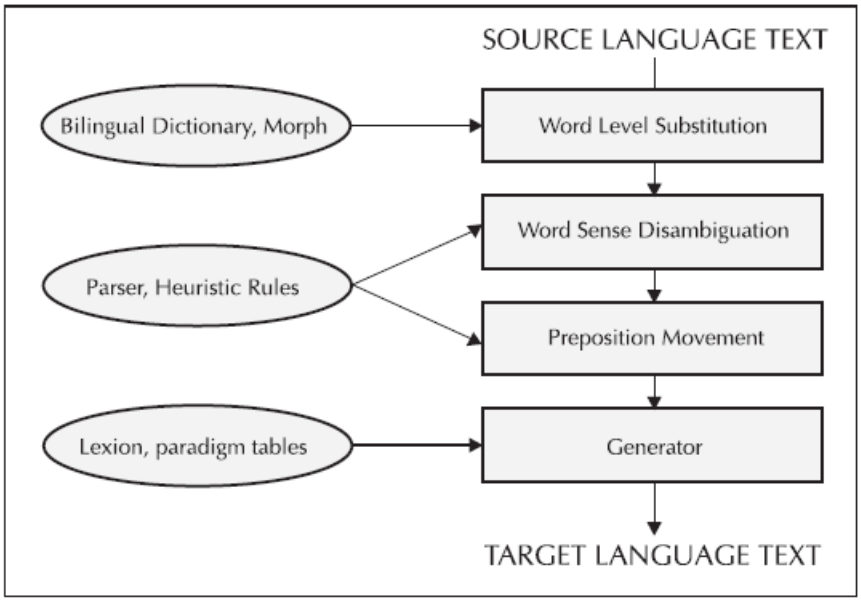

Figure 1 –The architecture of “core” anusaaraka

Architecture of the Anusaaraka system:

The Anusaaraka system has two major components.

_ Core engine
_ User–cum-developer interface

'Core' engine is the main engine of anusaaraka. This engine produces the output in different layers making the process of Machine Translation transparent to the user.

The architecture of "core" anusaaraka is shown in Figure 1.
This architecture differs from the conventional architecture in three major ways:

1. The order of operations is reversed. In the new architecture there is initial word level substitution followed by use of other language resources that are less reliable, like POS taggers, parsers, etc.

2. A graphical user interface has been developed to display the spectrum of outputs. The user has flexibility to adjust the output as per his/her needs. There will be users of different kinds based on the level of sophistication required and skill in handling the tool.

3. Special "interfaces", which act as 'glue' have been developed for different parsers, which allow plugging in of different parsers thereby providing modularity.

Core Anusaaraka engine

The core anusaaraka engine has four major modules viz.
I. Word Level Substitution
II. Word Sense Disambiguation
III. Preposition placement
IV. Word Order generation

I.Word Level Substitution

At this level the 'gloss' of each source language word into the target language is provided. However, the Polysemous words (words having more than one related meaning) create problems. When there is no one-one mapping, it is not practical to list all the meanings. On the other hand, anusaaraka claims 'faithfulness' to the original text. Then how is the faithfulness guaranteed at word level substitution?

II.Word Sense Disambiguation (WSD)

English has a very rich source of systematic ambiguity. Majority of nouns in English can potentially be used as verbs. Therefore, the WSD task in case of English can be split into two classes:
(i) WSD across POS
(ii) WSD within POS

The POS taggers can help in WSD when the ambiguity is across POSs. For example: Consider the two sentences 'He chairs the session'. 'The chairs in this room are comfortable'. The POS taggers mark the words with appropriate POS tags. These taggers use certain heuristic rules, and hence may sometimes go wrong. The reported performances of these POS taggers vary between 95% to 97%. However, they are still useful, since they reduce the search space for meanings substantially.

III.Preposition Placement

English has prepositions whereas Hindi has postpositions. Hence, it is necessary to move the prepositions to proper positions in Hindi before substituting their meanings. While moving the prepositions from their English positions to the proper Hindi positions, \record of their movements must be stored, so that in case a need arises, they can be reverted back to their original position. Therefore, the transformations performed by this module, are also reversible.

IV. Word Order Generation

Hindi is a free word order language. Therefore, even the anusaaraka output in the previous layer makes sense to the Hindi reader. However, this output not being natural in Hindi, may not be enjoyed as much as the output with natural Hindi order. Additionally, it would not be treated as a translation. Therefore, in this module the attempt is to generate the correct Hindi word order.

Interface for different linguistic tools

The second major contribution of this architecture is the concept of 'interfaces'. Machine translation requires language resources such as POS taggers, morphological analyzers, and parsers. More than one kinds of each of these tools exist. Hence, it is wise to use these tools. However, there are problems.

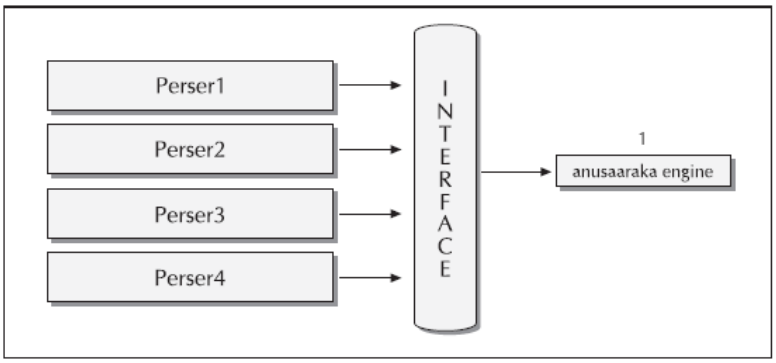

Figure 2 –Interfaces that map output of parsers to an intermediate form

As a machine translation system developer who is interested in the “usable” product one would like to plug-in different parsers and watch the performance. May be one would like to use combinations of them, or may like to vote among different parsers and choose the best parse out of them.
The Java/PYTHON based user interface has been developed to display the outputs produced by different layers of anusaaraka engine. The user interface provides a flexibility to control the display.

7.Multilingual Information Retrieval(MLIR)

Multilingual Information Retrieval (MLIR) is a subfield of Natural Language Processing (NLP) that focuses on retrieving relevant information from multilingual sources. It involves techniques and methodologies for searching, retrieving, and ranking documents or information in different languages.

The main goal of MLIR is to overcome language barriers and enable users to retrieve information from a diverse range of languages, even if they are not proficient in those languages. MLIR systems typically involve the following key components:

1. Multilingual Indexing: MLIR systems index documents from multiple languages to create a searchable collection. This process involves language-specific preprocessing techniques such as tokenization, stemming, and stop-word removal.
2. Cross-lingual Mapping: MLIR often involves creating mappings between different languages to establish connections and similarities. This can be achieved through techniques such as bilingual dictionaries, parallel corpora, or statistical models that learn cross-lingual word representations.
3. Query Translation: MLIR systems handle queries in one language and translate them into the languages of the indexed documents. Query translation methods include statistical machine translation, rule-based translation, or leveraging cross-lingual word embeddings.
4. Cross-lingual Retrieval Models: MLIR employs retrieval models that consider the multilingual nature of the indexed documents and queries. These models often combine language-specific relevance signals with cross-lingual information, such as document similarity or query expansion techniques.
5. Evaluation Metrics: MLIR systems are evaluated using metrics that account for the effectiveness of information retrieval across multiple languages. Common evaluation measures include mean average precision, precision at K, or cross-lingual variants of these metrics.

Challenges in MLIR include handling language-specific nuances, limited availability of resources for some languages, handling code-switching and mixed-language content, and scalability in indexing and retrieval for large multilingual collections.

MLIR finds applications in various domains such as cross-lingual search engines, multilingual digital libraries, e-commerce platforms, and information retrieval in multilingual social media content.

Researchers and practitioners in MLIR continue to explore advanced techniques, leveraging deep learning, neural networks, and transformer-based models to improve the effectiveness and efficiency of multilingual information retrieval systems.

8. Cross-Lingual Information Retrieval (CLIR)

CLIR stands for Cross-Language Information Retrieval, and it is a subfield of Natural Language Processing (NLP) that focuses on retrieving information across different languages. It involves the process of searching for and retrieving relevant documents or information in a target language, given a query expressed in a different source language.

The goal of CLIR is to bridge the language barrier and enable users to access information from different languages, even if they do not understand or speak those languages. It is particularly useful in multilingual and cross-cultural contexts, where people may need to search for information in languages they are not familiar with.

CLIR typically involves the following steps:

1. **Query Translation**: The user query, expressed in the source language, needs to be translated into the target language. This step can be challenging due to differences in grammar, vocabulary, and linguistic structure between languages.
2. **Document Indexing**: The documents in the target language need to be indexed to enable efficient retrieval. This typically involves extracting relevant features from the documents, such as keywords, named entities, or language-specific patterns.
3. **Retrieval**: The translated query is used to search the indexed documents, and retrieval algorithms rank the documents based on their relevance to the query. Various information retrieval techniques, such as vector space models or probabilistic models, can be applied here.
4. **Result Presentation**: The retrieved documents are presented to the user, often with additional processing to provide a summary, highlight relevant information, or support further exploration.

CLIR faces several challenges due to the inherent complexities of language translation, variations in language resources and structures, and the scarcity of parallel or bilingual data. Researchers in the field employ various techniques to address these challenges, including statistical machine translation, cross-lingual word embeddings, and leveraging multilingual resources such as dictionaries or parallel corpora.

CLIR has applications in areas such as multilingual search engines, digital libraries, cross-cultural communication, and global information access. It enables users to overcome language barriers and access information from diverse linguistic sources, fostering knowledge dissemination and collaboration across different language communities.

9. Evaluation in Information Retrieval

Evaluation in Information Retrieval (IR) in NLP refers to the process of assessing and measuring the effectiveness and performance of IR systems or models in retrieving relevant information in response to user queries. Evaluation plays a crucial role in assessing the quality of IR systems, comparing different approaches, and driving improvements in the field. There are several commonly used evaluation measures in IR:

1. **Precision:** Precision measures the proportion of retrieved documents that are relevant to a given query. It is calculated as the number of relevant documents retrieved divided by the total number of documents retrieved.
2. **Recall**: Recall measures the proportion of relevant documents that are retrieved out of all the relevant documents available in the collection. It is calculated as the number of relevant documents retrieved divided by the total number of relevant documents.
3. **F1 Score**: The F1 score combines precision and recall into a single metric, providing a balanced measure of system performance. It is the harmonic mean of precision and recall, calculated as 2 * (precision * recall) / (precision + recall).
4. **Mean Average Precision (MAP):** MAP is a widely used measure for ranked retrieval. It calculates the average precision across different recall levels for a set of queries. It considers the order in which documents are retrieved and rewards systems that retrieve relevant documents earlier in the ranked list.
5. **Normalized Discounted Cumulative Gain (NDCG):** NDCG is a measure that accounts for the relevance and rank of retrieved documents. It assigns higher scores to relevant documents that are ranked higher in the list. NDCG takes into account both precision and the position of relevant documents in the ranked list.
6. **Precision at K:** Precision at K measures the precision of the top-K retrieved documents. It considers only the first K documents and calculates the proportion of relevant documents among them.
7. **Mean Reciprocal Rank (MRR):** MRR measures the rank at which the first relevant document is retrieved. It calculates the reciprocal of the rank and takes the average across multiple queries.

These evaluation measures are used in experimental settings where a set of queries and relevant documents are predefined. The effectiveness of an IR system is evaluated by comparing its performance against a ground truth set of relevant documents. Additionally, evaluation may also involve user studies, where human assessors judge the relevance of retrieved documents based on their expertise or preferences.

It's important to note that evaluation in IR is an ongoing area of research, and different evaluation measures may be used depending on the specific task, dataset, or application. Researchers and practitioners continually work to develop new evaluation techniques that better reflect user needs and system performance in real-world scenarios.

10. Tools, Software and Resources

There are numerous tools, software libraries, and resources available for Natural Language Processing (NLP) that can assist with various NLP tasks. Here are some commonly used ones:

1. **NLTK (Natural Language Toolkit):** NLTK is a popular open-source library for NLP written in Python. It provides a wide range of functionalities and tools for tasks such as tokenization, stemming, tagging, parsing, and classification. It also offers access to corpora, lexical resources, and pre-trained models.
2. **spaCy**: spaCy is a Python library for advanced NLP tasks. It provides efficient tokenization, named entity recognition, part-of-speech tagging, dependency parsing, and lemmatization. spaCy focuses on performance and is known for its speed and ease of use.
3. **Gensim**: Gensim is a Python library for topic modeling, document similarity analysis, and other unsupervised NLP tasks. It offers implementations of popular algorithms like Latent Semantic Analysis (LSA), Latent Dirichlet Allocation (LDA), and Word2Vec.
4. **Stanford CoreNLP**: Stanford CoreNLP is a suite of NLP tools developed by Stanford University. It offers a wide range of capabilities, including tokenization, part-of-speech tagging, named entity recognition, dependency parsing, sentiment analysis, and coreference resolution. CoreNLP supports multiple languages and provides Java APIs along with wrappers for other programming languages.
5. **TensorFlow**: TensorFlow is an open-source deep learning framework that includes tools and libraries for NLP. It provides a high-level API called TensorFlow Hub, which offers pre-trained models for tasks like text classification, machine translation, and text generation. TensorFlow also supports the development of custom NLP models using deep learning architectures.
6. **PyTorch**: PyTorch is another popular deep learning framework that offers support for NLP. It provides tools for building and training neural networks, including modules for text classification, sequence labeling, and language generation. PyTorch also offers pre-trained models, such as BERT and GPT, for various NLP tasks.
7. **WordNet**: WordNet is a lexical database that organizes words into sets of synonyms called synsets. It also provides semantic relationships between words, such as hypernyms, hyponyms, and meronyms. WordNet is widely used for tasks like word sense disambiguation, lexical similarity, and semantic analysis.
8. **Word Embeddings**: Word embeddings are distributed representations of words in a continuous vector space. Pre-trained word embeddings, such as Word2Vec, GloVe, and FastText, are available for download and can be used to capture semantic relationships between words in NLP models.
9. **Universal Dependencies**: Universal Dependencies is a project that provides syntactic annotation standards for a large number of languages. It offers pre-annotated treebanks that represent the syntactic structure of sentences, which can be used for tasks like parsing and dependency analysis.
10. **BERT** (Bidirectional Encoder Representations from Transformers): BERT is a pre-trained deep learning model that has achieved state-of-the-art performance on various NLP tasks, including question answering, named entity recognition, and sentiment analysis. The original BERT model and its variations are available for fine-tuning and transfer learning.

These are just a few examples of the many tools, software libraries, and resources available for NLP. The choice of tools depends on the specific task, programming language preference, and the complexity of the project at hand. It's important to explore and experiment with different tools to find the ones that best suit your needs.