International Journal of Informatics and Communication Technology (IJ-ICT)
Vol. 14, No. 1, April 2025, pp. 1∼10
ISSN: 2252-8776, DOI: 10.11591/ijict.v14i1.pp1-10

A survey on novel approach to semantic computing for domain specific multi-lingual man-machine interaction

Anjali Bohra, Nemi Chand Barwar
Department of Computer Sciences and Engineering, MBM University, Jodhpur, India

Article Info

Article history:

Received Mar 5, 2024
Revised Jul 24, 2024
Accepted Sep 21, 2024

Keywords:

Deep learning
Karaka relations
Machine learning
Natural language processing
Panini grammar
Semantic computing
Semantic role labeling

ABSTRACT

Natural language processing (NLP) helps computational linguists to understand, process, and extract information from natural languages. Linguist Panini signifies 'information coding' in a language and explains that Karakas are semantico-syntactic relations between nouns and verbs that resemble participant roles of modern case grammar. Computational grammar maps vibhakti (inflections) of nominals and verbs to their participant roles. Karaka's theory extracts semantic roles in the sentences which act as intermediate steps for various NLP tasks. The survey shows that NLP seeks to bridge the gap for man-machine interaction. The work presents the impact of machine learning on natural language processing with changing trends from traditional to modern scenarios with Panini's classification scheme for semantic computing facilitating machine understanding. The study presents the significance of Karaka for semantic computing, methodologies for extracting semantic roles, and analysis of various deep learning-based language processing systems for applications like question answering. The survey covered around 50 research articles and 21 Karaka-based NLP systems performing multiple tasks like machine translation, question-answering systems, and text summaries using machine learning tools and frameworks. The work includes surveys from renowned journals, books, and relevant conferences, as well as descriptions of the latest trends and technologies in the machine learning domain.

Corresponding Author:

Anjali Bohra
Department of Computer Science and Engineering, MBM University
Jodhpur, Rajasthan, India
Email: anjalivb.phdcse@mbm.ac.in

1. INTRODUCTION

Artificial intelligence (AI) inculcates human abilities into machines by allowing learning through experience and adjusting to new inputs. Examples include computers playing cards, and digital assistants like Siri [1]. Computers are trained using AI technologies including machine learning, natural language processing (NLP), and computer vision to accomplish specific tasks [2]. Machines are trained through machine learning algorithms using data analysis and available patterns with minimal human interventions. Computers can communicate with humans in their language, through reading, identifying and classifying text, hearing and interpreting speech, and measuring sentiments using NLP techniques. Computer vision trains computers to analyze and understand the visual world by accurately identifying, and classifying objects, recognizing faces, processing live actions of a football game, and surpassing human visual abilities in many areas. Free multilingual machine translators developed by Google and Alexa developed by Amazon are prominent examples. AI technologies have transformed communication technology by shifting the data-driven paradigm to intelligence-driven endeavors. NLP helps machines to understand human language and behave as intelligently as humans by amalgamation of linguistics and computer science disciplines [3]. NLP analyzes different aspects of language like syntax, semantics, pragmatics, and morphology to transform linguistic knowledge into production-based algorithms for problem solution [4], [5]. Tasks include translation, relationship extraction, speech recognition, named-entity recognition, topic segmentation, sentiment analysis, Chatbots, and text Summarization. NLP tasks are performed in a sequence using a corpus and framework. A framework defines learning models using components that automatically understand, code, compute gradients, and perform parallel processing for optimized performance [6].

Basic approaches to NLP are distributional-based, frame-based, model-theoretical-based, and interactive-based learning [7]. Distributional-based approaches use statistical concepts focused on mathematical analysis of the content, including tasks like part-of-speech tagging, dependency parsing, and semantic relationships. Frame-based approaches consider frames as the standard for representing concepts. Model-theoretical-based approaches are semantic methods where the model defines the idea related to the concept and meaning of the sentence. Interactive learning approaches consider pragmatic concepts. Table 1 shows a list of designed language processing systems like sentiment analyzer, part of speech tagger, and emotion detection system through NLP methods and approaches. Understanding natural language has three stages of development: the rationalist stage, the empirical stage, and the deep learning stage.

Table 1. Langauge processing systems with NLP methods

S No	NLP systems	NLP methods with approaches
1	Sentiment analyser [8]	Topic as features (distributional approach)
2	Parts of speech taggers [9]	Rule based methods (distributional approach)
3	Chunking [10], [11]	Log-linear method/multi-label classification
4	Named entity recognition system [12], [13]	Statistical methods (distributional approach)
5	Emotion detection system [14]	Conditional random field method (model-theoretical approach)
6	Semantic role labelling system [15]	Semantic representation (frame-based-approach)
7	Event discovery system [16]	Latent semantic method (distributional approach)

The rationalist stage focuses on implementing Chomsky's rules for inducing reasoning and knowledge into NLP systems like ELIZA, and MARGIE. The empirical stage focused on implementing generalized concepts in machines through pattern recognition and generative models like HMM and IBM translation models. The current deep learning stage focuses on implementing a layered model to perform end-to-end learning for feature extraction. Dense representations of words, sentences, paragraphs, and documents are learned to capture both syntactic and semantic features. The numbers in word vector representation show the closeness of the encoded meaning with the specified concept [17]. NLP applications are hard and challenging as programming languages like Java and Python are required for man-machine interaction. These programming languages are structured and unambiguous while human languages are ambiguous as well as region adaptive [18]. The most difficult part of training computers using programming languages is handling lexical, referential, and syntax-level ambiguity with synonyms and hypernyms.

Semantic computing concentrates on understanding the meaning, interpretation, and relationships between words, phrases, and sentences through the grammar of a language to bridge the gap between people and computers [19]. It composes information based on meaning and vocabulary by implementing computing technologies (like artificial intelligence) through NLP, knowledge engineering, software engineering, and computer networks to extract, transform, and synthesize the content [20]–[22]. The key components of semantic analysis are lexical semantics, syntax word embedding, and vector space models. The study investigates the effect of deep learning for NLP which has achieved new benchmarks through distributed representation and semantic generalization of words. Contextual word embeddings in different contexts show different real-valued vector representations for the same word from a corpus [23], [24]. Word embedding of textual data is obtained using the embedding layer of Keras deep learning framework, Word2Vec or GloVe model, and bidirectional encoder representations from transformers (BERT) language model [25], [26]. Pre-trained embeddings have shown remarkable improvement in NLP tasks like speech recognition, syntactic parsing, text understanding and summarization, and question-answering systems [27]–[29].

Challenges in NLP: Despite major success in various NLP tasks like language modeling and machine translation, deep learning methods persist in lack of interpretability to interpret inter-sentential relations. More work is required in neural-symbolic representation of human knowledge [30]. Deficiency of knowledge, interpretability of models, and requirement of large datasets are the major challenges for NLP through deep learning. Reinforcement learning with inference, and knowledge-base lead to new learning paradigms [31]. Pragmatic interpretation is still an open area of research [32]. Word sense disambiguation, structural ambiguity, and co-reference resolution are challenges due to ambiguity and polysemy. Idiomatic expressions require contextual or cultural understanding. Lack of domain-specific knowledge misinterprets sentential relationships because different regions include unique terms and jargon that are unfamiliar to generalized language processing systems.

Research gaps in this paper include:

– Challenge is to develop a universal approach, large language model (LLM) based on Karak relations for mainstream Sanskrit, Hindi, and English language for NLP tasks like summarization, and translation.
– LLM suffers from hallucination which can be resolved through exact topic extraction techniques using semantic processing.
– The wide scope of research is open for multi-modal LLMs that combine text processing with audio, image, and video.

Work outline in this paper is brings together researchers from disciplines such as NLP, multimedia semantics, semantic Web, and pattern recognition to provide a single source for presenting the state of the technology to breakthroughs on the horizon. The introduction covers the history and development of machine learning's relevance to natural language processing with challenges to the field. Section 2 covers the background for NLP in semantic processing with the significance of Karak theory. The next section explains the methods followed with results and discussion. The last section summarizes the work with guidelines for future directions.

2. BACKGROUND: COMPONENTS OF A LANGUAGE

Linguistics considers language as a group of arbitrary vocal signs, governed by innate and universal rules (grammar) of the language. Grammar has two types: descriptive and Perspective grammar. Descriptive grammar defines a set of rules to formulate the speaker's grammar. Perspective grammar focuses on correctness in the language. A grammatical category is a class of units or features of a language indicating number, gender, degree, person, case, definiteness or indefiniteness, tense, aspect, mood, and agreement. Number is related to singular or plural concepts while gender is expressed by variation in personal pronouns or third person. Examples of grammatical genders are he, she, it (singular), I, we, and you (first and second form), and they (third person plural either common/neuter gender). Case shows the relationship of the noun phrase with verb and other noun phrases in a sentence like nominative case, genitive case, objective case, etc, and degree is shown by adjectives and adverbs. Tense grammatical category represents a time of an action and aspect defines a view of an event which can be perspective or imperative. Mood shows the speaker's attitude towards what he or she is talking about. Representing grammar (of a language) as mathematical expression is an intractable problem. Semantic networks, first-order logic, frames, and production systems are used for knowledge representation. Semantic networks describe the relation between an object and a class. Prolog programming language is based on a subset of first-order logic which is a declarative language for writing logic statements and proofs. The knowledge is converted into modular chunks using a frame-base approach while rules specifying patterns and actions are specified through a production system-based approach.

2.1. Karak theory

In linguistics, semantic analysis represents syntactic structures (words and phrases) with their language-independent meaning [33]. Linguist Panini defined a Karak-based approach for text and speech processing. He defined knowledge representation methodologies in his book 'Asthadhayayi' which are equivalent to current AI systems including meta-rules for coding AI software [34]. He developed a framework for universal grammar that can be applied to any natural language [35]–[37]. The framework is based on the concept of karma and morphosyntactic structures to extract semantic roles in a sentence. A semantic role describes the relation of a syntactic constituent (noun phrase) with a predicate (the verb or action) as an agent, patient, and instrument [38]. Paninian grammar processes sentences at four levels namely surface level (uttered sentence), bhakti level, Karaka level, and semantic level. Karakas specify relations between nominal and verbal root [39]. Following are the six Karakas specified by Panini according to their participation with the verb in a sentence: i) Karta: describes action of verb; ii) Karam: desired by the Karta Karak (subject); iii) Karana: act as instrument of the action performed by Karta; iv) Sampradaan: act as recipient of an action; v) Apaadaan: express detachment or comparision from a source; and vi) Adhikarana: describe place of action.

The Karaka-based approach is a template-based generation system which answer Karak-based questions with relevance to the case of noun phrases in the sentence. A noun or pronoun exists in eight forms in a sentence and therefore causes eight types of cases. Seven forms of vibhakti are nominative, accusative, instrument, dative, locative, gentive, and vocative [40]. Karaka relations are semantic-syntactic relations where Karta Karak acts as a nominative case, Karam Karak as an objective/accusative case, Karan Karak as an instrument, Sampradaan as a dative case, Apadan as an Ablative case and Sambandh is genetive/possessive case. Adhikaran Karak act as a locative case and Sambodhan as a vocative case [41]. Case is a property shared by all the languages of the world [42].

2.2. Methods of semantic processing

Semantic processing focuses on words to determine their significance in a phrase or a sentence [43]. Similarity measures are used to find the relevancy between the words [44]. Semantic processing methods decode the meaning within the text. The process starts with preprocessing and lexical analysis followed by parsing and syntactic analysis, semantic frame identification, and establishing mathematical representation of words through vector space models/embedding layers. Based on the required application suitable semantic analysis method is selected to extract the features. Finally, the system is evaluated for improving the performance using techniques such as semantic feature analysis, latent semantic analysis, and semantic content analysis [45].

– Semantic feature analysis emphasizes the representation of word features through feature selection (part of speech (POS) and morphological features), determining weights (through term frequency, inverse-term frequency, normalized term frequency, and global term weighting), and similarity measurement(through cosine/Jaccard similarity and euclidean distance).
– Latent semantic analysis captures the relationship of words with their context using statistical methods like reducing dimensionality and comparing semantic similarity. It is the mathematical method for extracting the meaning of words. The mathematics is to obtain parameters of any X rectangular tXp matrix of (r rank) terms and passage through decomposition into three matrices using singular value decomposition using (1).

  $$X = TSPT \tag{1}$$

  where T is txr matrix with orthonormal columns, P is pxr matrix with orthonormal columns and S is r x r diagonal matrix with sorted entries in descending order [46].
– Semantic content analysis identifies relationships between words and phrases using dependency parsing (graph-based parsing), thematic roles and case roles (reveals relationships between actions, participants, and objects), and identification of semantic frame.

3. LITERATURE REVIEW

Anusaarka, a language translation system based on paninian theory uses an interlingua-based approach which is an intermediate representation defined by verb, noun, and Karaka relations [47]–[49]. A rule-based Hindi lemmatizer that generates the rules for extracting suffixes from the given word [50], [51]. The government of India proposed a supervised learning-based Bengali root word extraction system using Paninian grammatical rules under the TDIL project [52]. Opinion classification system for Odia language using syntactic-semantic concept [53]. A list of dependency relations was prepared based on Panini's grammar which shows that relations represent well-defined semantics for extraction from the surface form of the word without any linguistic information [54]. Designed a paninian framework-based case marker error-resolver for Indian languages [55]. A Marathi Treebank was also designed based on Karak theory using Marathi corpus [56]. Natural language interface for databases was designed to process user queries(including logical operators, relational operators, and joining of tables for the Hindi language) by converting them into equivalent standard structured query language (SQL) query through computational Paninian grammatical framework [57]. Designed a constraint-based Parser for the Nepali language using Karak theory [58].

Table 2 summarizes Karak-based language processing systems performing machine translation tasks for Hindi, Sanskrit, and Malayalam languages, parsing of languages, and question-answering systems. The description includes their functioning, used methodology, datasets or corpus as well as evaluation results. These systems use definite words or sentences from specific corpus or datasets which are trained with features obtained from semantic processing. The systems are evaluated using precision and recall F-measure. All systems attained almost 75 to 95 percent accuracy in results.

Table 2. Karka-based language processing systems

S No.	System name	Description	Language	Method	Accuracy	Corpus/Dataset
1	Anusaarka	A language translation system	Kannada to Hindi, marathi, Bengali, and Telugu	Interlingua based method	92% approx.	30,000 words from Kannada dictionary and other language dictionaries
2	Hindi Lemmatizer [59]	Extracts suffixes from the root word	Hindi	Paradigm based method	0.89	2,500 words for Hindi dictionary
3	Root word extraction system	Extracts Bengali root word	Bengali	Rule based method	0.99	10,000 different inflected words from Bengali dictionary
4	Opinion classification system [60]	Classifies opinion of reviewers	Bengali	Topic based approach	0.7	Bengali newspaper available at http://www.ananda bazar.com/
5	Dependency-relations identification system [61]	lists dependency relations	Sanskrit	Production-based system	0.9	Bhagvat-Gita
6	Case-marker-errors identification system [62]	Identifies case marker errors for Indian languages committed by google machine translators	English to Urdu translation	Karak-vibhakti based dependency framework	Machine translation neural based 32% accurate and 21% phrase-based	500 English sentences
7	Sanskrit Karak analyzer [13]	Takes unicode Devnagri text and returns Karak analyzed text	Sanskrit	Rule based approach	84% accurate	31 Karaka, 72 vibhakti from sanskrit dictionary
8	Pilagiarism detection system [63]	Plagiarism detection system based on paninian framework	Malayalam	Machine learning approach		Online Malayalam newspapers
9	Verbframator [64]	Extracts verb frames for the given sentences	Marathi	Karaka based machine learning	Generate verb frames but require some human intervention	40,000 Marathi verbs from WordNet (subset of Indo-WordNet)
10	Question-answering system [65]	Generates questions in Hindi language	Hindi	Karak-based machine learning	5 pt Likert scale: 3.019, 3,336 syntactic and semantic mean	30 sentences from Hindi corpus
11	Question answering system [66]	Generate answers by comparing vibhakthi and POS tags of question words	Malayalam	Vibhathi and POS tagging based approach	Generate word level answers	Malayalam corpus
12	Semantic tagger and Karaka analyzer [51]	Perform tagging and identify Karaka	Hindi	rule-based approach	84% precise	Hindi corpus

Table 2. Karka-based language processing systems *(Continued)*

S No.	System name	Description	Language	Method	Accuracy	Corpus/Dataset
13	Text Clustering for a document [67]	Generate meaningful labels of the clusters	Punjabi	Karaka based machine learning approach	95% precise	Punjabi corpus
14	Generate semantic roles [68], [69]	Generic labels for the tokens of text	Malayalam	Karaka based machine learning approach		Malayalam corpus
15	Karakacross: sentiment analysis [70]	Extract sentiments related semantic roles	Different languages	Sentiment extraction using Karaka theory		Multi-lingual datasets
16	Text summarization system [71]	Perform single-document summarization	Malayalam	SRL based on Karaka theory	80 % precise	Online Malayalam repository
17	Cross-lingual study based on Karaka [72]	Impact of Karakas on congition	Sanskrit, Marathi, Kanada, and Telugu	Karaka based machine learning system	Karta and Karma mapped accurately	Sanskrit and Marathi language corpus
18	Case analyzer system [73]	Extract cases of Eastern Indo-Aryan languages	7 Indo-Aryan languages	Tradition and modern approach to study cognitive framework	80% accurate language-specific case relations	Corpus of Indo-Aryan languages
19	Question answering system [74]	Extraction of similarity features for classification in question answer (QA) selection	Hindi	Karaka based machine learning approach	Proper extraction of Karaka reduce needs role of pre-trained	Hindi corpus
20	Text summarization system [75]	Extractive summarization of a document	Malayalam	Machine learning based on Karaka theory	66% precise and 65% efficient in recall	Malayalam corpus
21	Question answering system [76]	Retrieval of answers for question answering	Hindi and Marathi	QA based on Karaka theory for Indic languages	80%, 60% precise for Hindi and marathi language	Hindi and Marathi corpus

4. METHOD

This review research on Karak-based multi-lingual language processing systems is relevant to answer questions related to the semantic interpretation of a language. Systematic literature review (SLR) has three parts: planning, construction, and reporting phase. The planning phase focuses on the need for a review accompanied by research questions. The construction phase selects primary studies and extracts data from those studies and the final stage disseminates results. The work explains the effectiveness of NLP in semantics to facilitate high-level programming languages (Prolog and Python) for computers.

5. RESULTS AND DISCUSSION

5.1. Results: Karak-based semantic rules in modern generative grammar

Karaka's theory is syntactic to the semantic formalization of language aspects. Case grammar described by fillmore regenerated the Paninian proposal in a modern linguistic context. He hypothesized human equivalent universal concepts for making judgments about the events or actions using the following answers to raised 5W (who/what/when/where/why) based questions [77], [78].

- Who is the initiator of the action?: Agent
- What is involved in the action?: Instrument (involved object)
- Who emphasis on the effect of the action?: Dative
- What is the result of the action?: Factitive (object)
- When and Where the event (or action) is oriented?: Locative
- Why the things are affected by the action?: Objective

Paninian-based Karak specifies answers to the questions for semantic interpretation of any natural language. Lexical, morphological, and syntactic features describe any language [79]. Lexico-syntactic features include POS tagging, morphological tagging includes root word, gender, number, person, and case, and syntactic features include head noun, chunk label, and dependency relation. Semantic role labeling is a semantic parsing technique widely used in question-answering systems or information extraction systems that assign semantic roles to syntactic constituents (arguments of predicate in a sentence). Karakas explained parsing Indian languages and creating Treebank for Hindi [80]. The Treebank dataset contains around four million annotated words divided into different annotations like parts-of-speech, syntactic, and semantic skeletons [81]. Sanchay is a free linguistic annotation tool for Indian languages published in a list of programs as part of education. Dependency-based formalism is incorporated for morphologically rich languages efforts have been incorporated for dependency-based formalism [82], [83]. Hyderabad dependency treebank (HyDT) for Hindi uses Karak relations to capture local semantics and labels relevant to the verb through dependency-based approach [84], [85].

5.2. Discussion: method of information coding in a language

Language has grammar (rules) for combining the words [86]. Languages use parsing to code the information. Semantic analysis helps in encoding the relations in a sentence. Grammar decides how the relations are coded in the language. Information can be summarized by answering 5W questions like who, what, when, where, and why. In machine translation, a given source is translated into the target language through 5Ws comprehensive [87]. Answering 5Ws generates domain-independent generic semantic roles. Paninian grammar signifies the minute observations regarding information coding in a language. Panini signifies information coding' in a language by answering three questions: where, which and how. Three aspects of questioning for extracting information coding in a language are: Where the information is coded? Which relations are coded in the sentence? And How the relations are coded?

A word can be tagged as nominal/verbal form according to the grammar. Tense and person morphologically inflect the word in a sentence. Each sentence is represented using alphabet letters and one sentence can be defined in terms of another exactly like the production rules of a Chomsky grammar [88]. Surface level (uttered sentence), vibhakti level, Karaka level, and semantic level are the four levels of text processing using the Paninian framework.

6. CONCLUSION

The paper presents a survey on paninian framework-based (Karak theory-based) language processing systems. It deals with a syntactico-semantic aspect of linguistics and the development stages of machine learning for NLP. The study suggests that syntactic-semantic concepts (semantic role labeling) have been leveraged through recent trends in machine learning algorithms and may benefit as a new paradigm of language-independent processing. The study explored a comprehensive work on the Paninian aspect of language processing with the latest trends in deep learning. However, in-depth studies are needed to get linguistic insights especially to understand speaker and listener communication. Researchers who want to utilize NLP for various purposes in their field can understand the overall technical status and the main technologies of NLP through this paper. Our study demonstrates that Karaka theory retains linguistic insights, which are more resilient than other semantic methods. The investigation opens a wide scope of research to unfold deeper linguistic aspects with feasible ways of unfolding cognition of Karaka in real-life man-machine interaction.

REFERENCES

[1] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[2] "SAS data and AI solutions," *Five AI technologies*. [Online]. Available: https://www.sas.com/en_in/insights/articles/analytics/five-ai-technologies.html.

[3] V. N. Gudivada and K. Arbabifard, "Open-source libraries, application frameworks, and workflow systems for NLP," in *Computational Analysis and Understanding of Natural Languages: Principles, Methods and Applications*, Series Handbook of Statistics, Elsevier, 2018, pp. 31–50.

[4] "Natural language processing: What it is? and how does it work?" [Online]. Available: https://monkeylearn.com/natural-language-processing.

[5] M. Raza, "Importance of NLP in data science," Medium, 2021. [Online]. Available: https://medium.com/theleanprogrammer/importance-of-nlp-in-data-science-3e2fd2424b2d.

[6] A. Shatnawi, G. Al-Bdour, R. Al-Qurran, and M. Al-Ayyoub, "A comparative study of open source deep learning frameworks," in *2018 9th International Conference on Information and Communication Systems, ICICS 2018*, 2018, vol. 2018-January, pp. 72–77, doi: 10.1109/IACS.2018.8355444.

[7] M. Yao, "Four approaches to natural language processing and understanding," *TopBots*, 2017. [Online]. Available: https://www.topbots.com/4-different-approaches-natural-language-processing-understanding.

[8] J. Yi, T. Nasukawa, R. Bunescu, and W. Niblack, "Sentiment analyzer: extracting sentiments about a given topic using natural language processing techniques," in *Proceedings - IEEE International Conference on Data Mining, ICDM*, 2003, pp. 427–434, doi: 10.1109/icdm.2003.1250948.

[9] N. Tapaswi and S. Jain, "Treebank based deep grammar acquisition and part-of-speech tagging for Sanskrit sentences," in *2012 CSI 6th International Conference on Software Engineering, CONSEG 2012*, 2012, pp. 1–4, doi: 10.1109/CONSEG.2012.6349476.

[10] F. Sha and F. Pereira, "Shallow parsing with conditional random fields," in *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, HLT-NAACL 2003*, 2003, pp. 213–220, doi: 10.3115/1073445.1073473.

[11] R. McDonald, K. Crammer, and F. Pereira, "Flexible text segmentation with structured multilabel classification," in *HLT/EMNLP 2005 - Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 2005, pp. 987–994, doi: 10.3115/1220575.1220699.

[12] A. Ritter, C. Sam, Mausam, and O. Etzioni, "Named entity recognition in tweets: an experimental study," in *EMNLP 2011 - Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 2011, pp. 1524–1534.

[13] S. K. Mishra and G. N. Jha, "Sanskrit Kāraka analyzer for machine translation," in *SPLASH proceeding of iSTRANS*, 2007, pp. 224–225.

[14] A. Ilyas, K. Shahzad, and M. Kamran Malik, "Emotion detection in code-mixed Roman Urdu-English text," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 2, pp. 1–28, 2023.

[15] P. Martha, "The proposition bank: An annotated corpus of semantic roles," *Computational Linguistics*, vol. 31, no. 1, pp. 71–105, 2005.

[16] E. Benson, A. Haghighi, and R. Barzilay, "Event discovery in social media feeds," in *ACL-HLT 2011 - Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 2011, vol. 1, pp. 389–398.

[17] A. Bohra and N. C. Barwar, "Performance evaluation of word representation techniques using deep learning methods," in *Proceedings of the 2020 International Conference on Computing, Communication and Security, ICCCS 2020*, 2020, pp. 1–7, doi: 10.1109/ICCCS49678.2020.9277190.

[18] A. Bharati, R. Sangal, and V. Chaitanya, "Natural language processing, complexity theory and logic," in *Foundations of Software Technology and Theoretical Computer Science: Tenth Conference, Bangalore, India December 17–19, 1990 Proceedings 10*, 1990, vol. 472 LNCS, pp. 410–420, doi: 10.1007/3-540-53487-3_60.

[19] L. Wang, H. Wang, and S. Yu, "Semantic computing and language knowledge bases," in *IOP Conference Series: Materials Science and Engineering*, 2017, vol. 242, no. 1, p. 12109, doi: 10.1088/1757-899X/242/1/012109.

[20] P. Sheu, "Semantic computing and cognitive computing/informatics," in *2017 IEEE 16th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC)*, 2017, pp. 4–4, doi: 10.1109/icci-cc.2017.8109801.

[21] P. C.-Y. Sheu, *Semantic computing*. H. Yu, C. V. Ramamoorthy, A. K. Joshi, L. A. Zadeh, Ed., Wiley Online Library, 2010.

[22] S.-U. Park, "Analysis of the status of natural language processing technology based on deep learning," *The Journal of Bigdata*, vol. 6, no. 1, pp. 63–81, 2021.

[23] S. Tekir and Y. Bastanlar, "Deep learning: exemplar studies in natural language processing and computer vision," *Data Mining - Methods, Applications and Systems*, p. 1, 2020, doi: 10.5772/intechopen.91813.

[24] J. Browniee, "Deep learning for natural language processing," in *Machine Learning Mystery*, 2017.

[25] N. Agarwal, "The ultimate guide to different word embedding techniques in NLP," *KDnuggets*, 2022. https://www.kdnuggets.com/2021/11/guide-word-embedding-techniques-nlp.html.

[26] D. Suhartono, A. P. Gema, S. Winton, T. David, M. I. Fanany, and A. M. Arymurthy, "Argument annotation and analysis using deep learning with attention mechanism in Bahasa Indonesia," *Journal of Big Data*, vol. 7, no. 1, pp. 1–18, 2020, doi: 10.1186/s40537-020-00364-z.

[27] T. Katte, "Recurrent neural network and its various architecture types," *International Journal of Research and Scientific Innovation (IJRSI)*, vol. 5, no. 3, pp. 124–129, 2018.

[28] R. Salakhutdinov and G. Hinton, "Deep Boltzmann machines," in *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, 2009, vol. 5, pp. 448–455.

[29] N. Sethi, A. Dev, and P. Bansal, "A novel neural machine translation approach for low-resource Sanskrit-Hindi language pair," *ACM Transactions on Asian and Low-Resource Language Information Processing*, Apr. 2023, doi: 10.1145/3591207.

[30] L. Ferrone and F. M. Zanzotto, "Symbolic, distributed, and distributional representations for natural language processing in the era of deep learning: a survey," *Frontiers in Robotics and AI*, vol. 6, p. 153, 2020, doi: 10.3389/frobt.2019.00153.

[31] H. Li, "Deep learning for natural language processing: advantages and challenges," *National Science Review*, vol. 5, no. 1, pp. 24–26, 2018, doi: 10.1093/nsr/nwx110.

[32] A. Bharati, V. Chaitanya, R. Sangal, and B. Gillon, *Natural language processing: A Paninian perspective*. Prentice Hall India Learning Private Limited, 1995.

[33] S. Selot, N. Tripathi, and A. S. Zadgaonkar, "Neural network model for semantic analysis of Sanskrit text," *International Journal of Natural Computing Research*, vol. 7, no. 1, pp. 1–14, 2018, doi: 10.4018/ijncr.2018010101.

[34] S. D. Joshi, "Syntactic and semantic devices in the Astādhyāyī of Pāṇini," *Journal of Indian Philosophy*, vol. 29, no. 1–2, pp. 155–167, 2001, doi: 10.1023/a:1017559527007.

[35] S. C. Kak, "The Paninian approach to natural language processing," *International Journal of Approximate Reasoning*, vol. 1, no. 1, pp. 117–130, 1987, doi: 10.1016/0888-613X(87)90007-7.

[36] P. R. Chaudhari, P. C. Gangurde, and N. L. Kulkarni, "Study of methodologies for utilizing Sanskrit in computational linguistics," *International Journal of Electronics, Communication and Soft Computing Science & Engineering*, p. 1, 2015.

[37] D. Jha, R. Jha, and V. Varshney, "Natural language processing and Sanskrit," *International Journal of Computer Engineering & Technology*, vol. 5, no. 10, pp. 57–63, 2014.

[38] Amita and A. Jangra, "An annotation scheme for English language using Paninian framework," *IJISET- International Journal of Innovative Science, Engineering & Technology*, vol. 2, no. 1, pp. 616–619, 2015.

[39] A. Bharati and R. Sangal, "Computational Paninian grammar framework," in *Supertagging: Using complex lexical descriptions in natural language processing*, 2010, pp. 355–370.

[40] S. Chatterji, T. M. Sarkar, S. Sarkar, and J. Chakraborty, "Karak relations in Bengali," in *Proceedings of 31st All-India Conference of Linguists (AICL 2009)*, 2009, pp. 33–36.

[41] B. V. S. Perraju, "Algorithmic aspects of natural language parsing using Paninian framework," M. Tech thesis, Department of Computer Science and Engineering, IIT Kanpur, 1992.

[42] K. Alla, R. Prasad, and R. R. Kandula, "A new approach to Hindi text steganography using Hindi Karak Kriyaye," *Journal of Information Assurance & Security*, vol. 6, no. 6, 2011.

[43] A. Badawi, "The effectiveness of natural language processing (NLP) as a processing solution and semantic improvement," *International Journal of Economic, Technology and Social Sciences*, vol. 2, no. 1, pp. 36–44, 2021, doi: 10.53695/injects.v2i1.194.

[44] S. Li, J. Zhang, X. Huang, S. Bai, and Q. Liu, "Semantic computation in a Chinese question-answering system," *Journal of Computer Science and Technology*, vol. 17, no. 6, pp. 933–939, 2002, doi: 10.1007/BF02960786.

[45] S. A. Salloum, R. Khan, and K. Shaalan, "A survey of semantic analysis approaches," in *Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2020)*, 2020, vol. 1153 AISC, pp. 61–70, doi: 10.1007/978-3-030-44289-7_6.

[46] T. Landauer and S. Dumais, "Latent semantic analysis," *Scholarpedia*, vol. 3, no. 11, p. 4356, 2008, doi: 10.4249/scholarpedia.4356. Available: http://www.scholarpedia.org/article/Latentsemanticanalysis?ref=https://githubhelp.com

[47] A. Bharati, M. Bhatia, V. Chaitanya, and R. Sangal, "Paninian grammar framework applied to English," Department of Computer Science and Engineering, Indian Institute of Technology, Kanpur, 1996.

[48] S. Mittal, "A review of English to Indian language translator: Anusaaraka," *IJCA Proceedings on International Conference on Advances in Computer Engineering and Applications*, vol. ICACEA, no. 1, pp. 1–6, 2014.

[49] A. P. Kulkarni, "Design and architecture of 'Anusaaraka'-An approach to machine translation," *Satyam Technical Review*, vol. 3, pp. 57–64, 2003.

[50] S. Paul, M. Tandon, N. Joshi, and I. Mathur, "Design of a rule based Hindi lemmatizer," in *Proceedings of Third International Workshop on Artificial Intelligence, Soft Computing and Applications, Chennai, India*, 2013, vol. 2, pp. 67–74, doi: 10.5121/csit.2013.3408.

[51] P. Katyayan and N. Joshi, "Development of automatic rule-based semantic tagger and Karaka analyzer for Hindi," *Transactions on Asian and Low-Resource Language Information Processing*, vol. 21, no. 2, pp. 1–25, 2021, doi: 10.1145/3479155.

[52] A. Das, T. Halder, and D. Saha, "Automatic extraction of Bengali root verbs using Paninian grammar," in *RTEICT 2017 - 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, Proceedings*, 2017, vol. 2018-January, pp. 953–956, doi: 10.1109/RTEICT.2017.8256739.

[53] B. K. Mishra and R. Sahoo, "A hybrid knowledge mining approach to develop a system framework for Odia language text processing," *Materials Today: Proceedings*, vol. 5, no. 1, pp. 1335–1340, 2018, doi: 10.1016/j.matpr.2017.11.219.

[54] J. Bronkhorst, "The role of meanings in Panini's grammar," *Indian Linguistics*, vol. 40, pp. 146–157, 1979.

[55] S. Bhate and S. Kak, "Pāṇini's grammar and computer science," *Annals of the Bhandarkar Oriental Research Institute*, vol. 72, no. 1/4, pp. 79–94, 1991.

[56] P. Lahoti, N. Mittal, and G. Singh, "A survey on NLP resources, tools, and techniques for Marathi language processing," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 2, pp. 1–34, 2022, doi: 10.1145/3548457.

[57] A. Kataria and R. Nath, "Natural language interface for databases in Hindi based on Karaka theory," *International Journal of Computer Applications*, vol. 122, no. 7, pp. 39–43, 2015, doi: 10.5120/21716-4841.

[58] A. Yajnik and D. Sharma, "Parsing techniques using Paninian framework on Nepali language," *DJ Journal of Engineering and Applied Mathematics*, vol. 1, no. 1, pp. 23–28, 2015, doi: 10.18831/djmaths.org/2015011004.

[59] N. Kanuparthi, A. Inumella, and D. M. Sharma, "Hindi derivational morphological analyzer," in *Proceedings of the Twelfth Meeting of the Special Interest Group on Computational Morphology and Phonology*, pp. 10–16, 2012.

[60] A. Das, "Opinion extraction and summarization from text documents in Bengali," Department of Computer Science & Engineering, Jadavpur University, 2011.

[61] V. N. Misra, *The descriptive technique of Paṇini: an introduction*, vol. 18. Walter de Gruyter GmbH & Co KG, 2017.

[62] S. Muzaffar, P. Behera, and G. N. Jha, "A Pāniniān framework for analyzing case marker errors in English-Urdu machine translation," *Procedia Computer Science*, vol. 96, pp. 502–510, 2016, doi: 10.1016/j.procs.2016.08.117.
S. K. Mishra and G. N. Jha, "Sanskrit Kāraka analyzer for machine translation," in *SPLASH proceeding of iSTRANS*, 2007, pp. 224–225.

[63] L. Sindhu, "An integrated approach for plagiarism detection in Malayalam documents," Cochin University of Science and Technology, 2017.

[64] H. Redkar *et al.*, "Verbframator: Semi-automatic verb frame annotator tool with special reference to Marathi," in *Proceedings of the 13th International Conference on Natural Language Processing*, 2016, pp. 299–304.

[65] K. Anuranjana, V. A. Rao, and R. Mamidi, "Hindi question generation using dependency structures," arXiv preprint arXiv:1906.08570, 2019, [Online]. Available: http://arxiv.org/abs/1906.08570.

[66] S. M. Archana, N. Vahab, R. Thankappan, and C. Raseek, "A rule based question answering system in Malayalam corpus using Vibhakthi and POS tag analysis," *Procedia Technology*, vol. 24, pp. 1534–1541, 2016, doi: 10.1016/j.protcy.2016.05.124.
P. Katyayan and N. Joshi, "Development of automatic rule-based semantic tagger and Karaka analyzer for Hindi," *Transactions on Asian and Low-Resource Language Information Processing*, vol. 21, no. 2, pp. 1–25, 2021, doi: 10.1145/3479155.

[67] S. Sharma and V. Gupta, "Punjabi documents clustering system," *Journal of Emerging Technologies in Web Intelligence*, vol. 5, no. 2, pp. 171–187, 2013, doi: 10.4304/jetwi.5.2.171-187.

[68] J. P. Jayan, J. S. Kumar, and T. Amudha, "Semantic role identification for Malayalam using machine learning approaches," *Innovations in Systems and Software Engineering*, pp. 1–7, 2023, doi: 10.1007/s11334-022-00496-w.

[69] K. T. Radhika and P. C. R. Raj, "Semantic role extraction and general concept understanding in Malayalam using Paninian grammar," *International Journal of Engineering Research and Development*, vol. 9, no. 3, pp. 28–33, 2013.

[70] D. Rai, "KarakACROSS: a Karaka-based approach to cross-lingual sentiment analysis," *International Journal of Literature and Languages*, vol. 3, no. 09, pp. 1–4, 2023, doi: 10.37547/ijll/Volume03Issue09-01 01-04 0.

[71] M. Rahul Raj, R. P. Haroon, and N. V. Sobhana, "A novel extractive text summarization system with self-organizing map clustering and entity recognition," *Sadhana - Academy Proceedings in Engineering Sciences*, vol. 45, no. 1, p. 32, 2020, doi: 10.1007/s12046-019-1248-0.

[72] J. A. Gajjam, "Taking a road not travelled: A cross-lingual study of Kāraka-cognition," *Language in India*, vol. 21, no. 5, pp. 1–24, 2021.

[73] B. Lahiri, *The case system of Eastern Indo-Aryan languages: a typological overview*. Routledge India, 2021.

[74] D. A. Verma, R. S. Joshi, S. A. Joshi, and O. K. Susladkar, "Study of similarity measures as features in classification for answer sentence selection task in Hindi question answering: language-specific vs other measures," in *Proceedings of the 35th Pacific Asia Conference on Language, Information and Computation, PACLIC 2021*, 2021, pp. 715–724.

[75] J. P. Jayan and V. Govindaru, "Automatic summarization of Malayalam documents using text extraction methods," in *SCRS Conference Proceedings on Intelligent Systems*, 2021, pp. 443–457, doi: 10.52458/978-93-91842-08-6-42.

[76] D. Verma, R. Joshi, A. Shivani, and R. Gupta, "Karaka-based answer retrieval for question answering in Indic languages," in *International Conference Recent Advances in Natural Language Processing, RANLP*, 2023, pp. 1216–1224, doi: 10.26615/978-954-452-092-2_129.

[77] A. Vaidya, S. Husain, P. Mannem, and D. M. Sharma, "A Karaka based annotation scheme for English," in *Computational Linguistics and Intelligent Text Processing: 10th International Conference, CICLing 2009, Mexico City, Mexico, March 1-7, 2009. Proceedings 10*, 2009, vol. 5449 LNCS, pp. 41–52, doi: 10.1007/978-3-642-00382-0_4.

[78] A. Bharati and R. Sangal, "Parsing free word order languages in the Paninian framework," in *31st Annual Meeting of the Association for Computational Linguistics*, 1993, vol. 1993-June, pp. 105–111, doi: 10.3115/981574.981589.

[79] R. Briggs, "Knowledge representation in Sanskrit and artificial intelligence," *AI Magazine*, vol. 6, no. 1, p. 32, 1985, [Online]. Available: http://www.aaai.org/ojs/index.php/aimagazine/article/view/466/402.

[80] S. Singh and T. J. Siddiqui, "Role of Karaka relations in Hindi word sense disambiguation," *Journal of Information Technology Research (JITR)*, vol. 8, no. 3, pp. 21–42, 2015, doi: 10.4018/JITR.2015070102.

[81] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mechanical Systems and Signal Processing*, vol. 115, pp. 213–237, 2019, doi: 10.1016/j.ymssp.2018.05.050.

[82] S. K. Karn, "Case system in Maithili: from semantic perspective," *Nepalese Linguistics*, vol. 23, p. 126, 2008.

[83] "Natural language processing tutorial," tutorialspoint, [Online]. Available: https://www.tutorialspoint.com/natural_language_processing/index.htm.

[84] O. Rambow, B. Dorr, K. Kipper, I. Kučerová, and M. Palmer, "Automatically deriving tectogrammatical labels from other resources: a comparison of semantic labels across frameworks," *Prague Bulletin of Mathematical Linguistics*, vol. 79–80, pp. 23–35, 2003.

[85] P. J. Antony and K. P. Soman, "Computational morphology and natural language parsing for Indian languages: a literature survey," *International Journal of Scientific & Engineering Research*, vol. 3, no. 3, pp. 136–146, 2012.

[86] A. Bharati and A. Kulkarni, "Information coding in a language: some insights from Paninian grammar," 2008.

[87] K. Parton *et al.*, "Who, what, when, where, why? Comparing multiple approaches to the cross-lingual 5W task," in *ACL-IJCNLP 2009 - Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and 4th International Joint Conference on Natural Language Processing of the AFNLP, Proceedings of the Conference*, 2009, pp. 423–431.

[88] A. Bharati and R. Sangal, "A Karaka based approach to parsing of Indian languages," in *COLING 1990 Volume 3: Papers presented to the 13th International Conference on Computational Linguistics*, 1990, pp. 25–29, doi: 10.3115/991146.991151.

BIOGRAPHIES OF AUTHORS

Anjali Bohra received her first degree Bachelor of Engineering in Computer Science and Engineering from Mody College of Engineering and Technology, Rajasthan University, Laxmangarh, Rajasthan 2002. She has also attained Master degree in Computer Science and Engineering from MBM Engineering College, Jai Narain Vyas University, Jodhpur, Rajasthan in 2012. Currently a Ph.D. scholar and her research interests focus on natural language processing, artificial intelligence, machine learning, and deep learning. She can be contacted at email: anjalivb.phdcse@mbm.ac.in.

Nemi Chand Barwar has B.E. in Computer Technology from MANIT Bhopal, M.E. in Digital Communication, and a Ph.D. from MBM Engineering College, Jodhpur. He works as a Professor at, the Department of Computer Science & Engineering, MBM University, Jodhpur. He has experience of over 30 years in the field of teaching and research. He has published more than 60 research papers in national and international conferences and journals. He is supervising the Ph.D. research program in computer science and engineering discipline as well as in Information. His research and teaching interests are computer networking, WSN, MANET/VANET, IoT, big data analytics, VoD, P2P networks, and machine learning. He had organized 10 national conferences and short-term courses sponsored by AICTE/UGC/DST. He is a life member of ISTE, IEI. He can be contacted at email: nemi.cse@mbm.ac.in.