ISSN: 2349-6002

पाणिनीय व्याकरण और प्राकृतिक भाषा प्रसंस्करण: अष्टाध्यायी का संगणकीय विश्लेषण ।

जल्पा एस. मुंजपरा।
M.Phil, NET, GSET, विद्यावारिधी शोध छात्रा गुजरात विश्वविद्यालय।
doi.org/10.64643/IJIRTV12I8-191665-459

सारांश- यह शोध पत्र प्राचीन भारतीय वैयाकरण पाणिनि द्वारा रचित 'अष्टाध्यायी' की संरचना और आधुनिक संगणकीय भाषाविज्ञान (Computational Linguistics) के बीच विद्यमान गहन संबंधों का एक विस्तृत और विश्लेषणात्मक विवरण प्रस्तुत करता है। पाणिनि की अष्टाध्यायी, जिसे लगभग ५००-३५० ईसा पूर्व के बीच निर्मित माना जाता है, विश्व की पहली औपचारिक भाषा प्रणाली (Formal Language System) के रूप में पहचानी जाती है, जो वर्तमान के प्राकृतिक भाषा प्रसंस्करण (NLP), प्राकृतिक भाषा बोध (NLU) और प्राकृतिक भाषा जनन (NLG) के मौलिक सिद्धांतों को आधार प्रदान करती है । इस शोध का मुख्य उद्देश्य यह प्रदर्शित करना है कि कैसे पाणिनि के लगभग ४,००० सूत्र एक स्व-निहित, नियम-आधारित इंजन के रूप में कार्य करते हैं, जो आधुनिक प्रोग्रामिंग तर्क और एल्गोरिथम सोच के साथ अद्भुत समानता रखते हैं । विश्लेषण के दौरान, 'कारक' (Karaka) सिद्धांत को सिमेंटिक विश्लेषण के एक सशक्त उपकरण के रूप में और 'प्रक्रिया' (Prakriya) को एक अनुक्रमिक जनरेटिव मॉडल के रूप में देखा गया है, जो 'बैकस-नौर फॉर्म' (BNF) और 'चोमस्की पदानुक्रम' (Chomsky Hierarchy) के पूर्ववर्ती सिद्ध होते हैं । इसके अतिरिक्त, यह पत्र आधुनिक हाइब्रिड मॉडल (Neural + Rule-based) में पाणिनीय नियमों के अनुप्रयोग और मशीनी अनुवाद में उनकी प्रभावशीलता की भी समीक्षा करता है।

मुख्य शब्द - पाणिनि, अष्टाध्यायी, प्राकृतिक भाषा प्रसंस्करण (NLP), कारक सिद्धांत, संगणकीय भाषाविज्ञान, औपचारिक भाषा सिद्धांत, मशीनी अनुवाद, ज्ञान निरूपण (Knowledge Representation), एल्गोरिथम, व्याकरणिक जनन।

१. भूमिका: पाणिनि[i] और सूचना सिद्धांत [ii]का ऐतिहासिक परिप्रेक्ष्य

प्राचीन भारत के ज्ञान-विज्ञान की परंपरा में महर्षि पाणिनि का योगदान केवल एक भाषा के व्याकरण तक सीमित नहीं है, बल्कि उन्होंने मानवीय बुद्धि के सबसे उत्कृष्ट स्मारकों में से एक, 'अष्टाध्यायी' का निर्माण किया । गान्धार(पाकिस्तान - रावलपिंडी) क्षेत्र में जन्मे पाणिनि ने संस्कृत भाषा को एक अत्यंत परिष्कृत और वैज्ञानिक ढांचा प्रदान किया, जिसे 'संस्कारित' या परिष्कृत भाषा कहा गया । आधुनिक संगणक विज्ञान के उदय के साथ, विशेष रूप से २०वीं शताब्दी के उत्तरार्ध में, भाषाविदों और कंप्यूटर वैज्ञानिकों ने यह अनुभव किया कि पाणिनि की पद्धति और आधुनिक कोडिंग सिद्धांतों के बीच एक संरचनात्मक सेतु विद्यमान है ।

पाणिनीय व्याकरण की मौलिकता इसकी 'जेनरेटिव' (Generative) प्रकृति में निहित है। नोम चोमस्की ने स्वयं स्वीकार किया कि आधुनिक अर्थों में पहला जेनरेटिव व्याकरण पाणिनि का ही था । अष्टाध्यायी एक ऐसी मशीन की तरह कार्य करती है जिसमें 'धातु' (verbal roots) और 'प्रातिपदिक' (nominal stems) को इनपुट के रूप में डाला जाता है और नियमों की एक श्रृंखला के माध्यम से वे व्याकरणिक रूप से शुद्ध पदों (words) में परिवर्तित हो जाते हैं । यह प्रक्रिया वर्तमान के 'नेचुरल लैंग्वेज जनरेशन' (NLG) की मूलभूत अवधारणा के समान है, जहाँ डेटा से अर्थपूर्ण वाक्यों का निर्माण किया जाता है ।

संगणकीय भाषाविज्ञान के क्षेत्र में पाणिनि की प्रासंगिकता का एक अन्य आयाम असंदिग्धता'अनअम्बिग्युइटी'[iii] (Unambiguity) है। कंप्यूटर को भाषाओं को समझने के लिए स्पष्ट और तर्कसंगत नियमों की आवश्यकता होती है। पाणिनि ने अपनी धातु-प्रत्यय व्यवस्था और सूत्रों के माध्यम से एक ऐसा तंत्र विकसित किया जो किसी भी प्रकार के संशय के लिए स्थान नहीं छोड़ता । इसी कारण से, १९८० के दशक में नासा के वैज्ञानिक रिक ब्रिग्स[iv] ने संस्कृत को कृत्रिम बुद्धिमत्ता (AI) के लिए एक संभावित मॉडल के रूप में प्रस्तावित किया था ।

२. अष्टाध्यायी की वास्तुकला: एक जटिल सॉफ्टवेयर डिजाइन

अष्टाध्यायी की संरचना किसी आधुनिक सॉफ्टवेयर

आर्किटेक्चर की तरह अत्यंत व्यवस्थित और श्रेणीबद्ध है। इसमें चार मुख्य घटक शामिल हैं जो मिलकर एक संपूर्ण भाषाई पारिस्थितिकी तंत्र का निर्माण करते हैं । इन घटकों के बीच का अंतर्संबंध डेटाबेस प्रबंधन और एल्गोरिथम प्रसंस्करण का एक प्राचीन उदाहरण प्रस्तुत करता है।

अष्टाध्यायी के घटक	संगणकीय समकक्ष	कार्य और भूमिका
सूत्रपाठ (Sutra-patha)	प्रोग्राम कोड / एल्गोरिथम (Program Code)	४,००० संक्षिप्त नियमों का संग्रह जो व्याकरणिक प्रक्रिया को संचालित करते हैं।
शिवसूत्र (Shiva-sutras)	वर्णमाला / डेटा संरचना (Data Structure)	ध्वनियों का विन्यास जो प्रत्याहार (abbreviations) बनाने में सहायक होता है।
धातुपाठ(Dhatu-patha)	क्रिया डेटाबेस (Verbal Database)	लगभग २,००० क्रिया मूलों की सूची जो अर्थ और गुणों के साथ संचित हैं।
गणपाठ (Gana-patha)	लेक्सिकल श्रेणियाँ / अपवाद सूची (Lexical Lists)	विशेष शब्दों के समूह जो विशिष्ट नियमों के अधीन होते हैं।

२.१ शिवसूत्र और प्रत्याहार: डेटा कम्प्रेशन की तकनीक :

पाणिनीय तंत्र का आधार 'शिवसूत्र' हैं, जो ध्वनियों को एक विशेष क्रम में प्रस्तुत करते हैं। इन १४ सूत्रों के अंत में एक 'इत्' (It) वर्ण होता है, जो केवल एक मार्कर या फ्लैग की तरह कार्य करता है । इन 'इत्' वर्णों के उपयोग से पाणिनि 'प्रत्याहार' का निर्माण करते हैं, जो ध्वनियों के बड़े समूहों को संक्षिप्त रूप में दर्शाने की एक विधि है। उदाहरण के लिए, 'अच्' कहने से सभी स्वर और 'हल्' कहने से सभी व्यंजन सूचित होते हैं ।

यह तकनीक आधुनिक कंप्यूटर विज्ञान में 'डेटा कम्प्रेशन' (Data Compression) और 'रेगुलर एक्सप्रेशन' (Regex) के समान है । जहाँ एक वैयाकरण को सैकड़ों ध्वनियों को बार-बार सूचीबद्ध नहीं करना पड़ता, वहीं कंप्यूटर प्रोग्रामिंग में भी हम 'क्लासेस' (Classes) का उपयोग इसी संक्षिप्तता के लिए करते हैं। पाणिनि का 'लाघव' (संक्षिप्तता) सिद्धांत यह सुनिश्चित करता है कि व्याकरण का 'मेमोरी फुटप्रिंट' न्यूनतम रहे, जो कि आधुनिक एल्गोरिथम अनुकूलन (Algorithm Optimization) का मुख्य लक्ष्य है ।

२.२ धातुपाठ और गणपाठ: ऑन्कोलॉजी और डेटा डिक्शनरी :

अष्टाध्यायी के सहायक ग्रंथ 'धातुपाठ' और 'गणपाठ' आधुनिक एआई में उपयोग होने वाली 'ऑन्कोलॉजी' (Ontology) और 'डेटा डिक्शनरी' के पूर्वज कहे जा सकते हैं । धातुपाठ में क्रियाओं को न केवल उनकी ध्वनि के आधार पर, बल्कि उनके अर्थगत गुणों (जैसे आत्मनेपद/परस्मैपद) के आधार पर भी वर्गीकृत किया गया है । यह वर्गीकरण आधुनिक एनएलपी प्रणालियों में 'फीचर इंजीनियरिंग' (Feature Engineering) के समान है, जहाँ हम शब्दों को उनके व्याकरणिक व्यवहार के आधार पर टैग करते हैं। गणपाठ उन शब्दों का समूह है जो सामान्य नियमों के अपवाद हैं, जिससे प्रोग्रामिंग में 'एक्सेप्शन हैंडलिंग' (Exception Handling) जैसी स्पष्टता बनी रहती है ।

३. औपचारिक भाषा सिद्धांत[v] और संगणकीय तर्क :

कंप्यूटर विज्ञान में 'औपचारिक भाषा सिद्धांत' (Formal Language Theory) यह निर्धारित करता है कि कोई भाषा कितनी जटिल है और मशीन उसे कितनी आसानी से संसाधित कर सकती है। पाणिनि के सूत्रों की प्रकृति 'प्रोडक्शन रूल्स' (Production Rules) की तरह है, जो एक प्रतीक को दूसरे में बदलने का निर्देश देते हैं । शोधकर्ताओं ने यह सिद्ध किया है कि पाणिनि की सूत्र शैली 'बैकस-नौर फॉर्म' (BNF) के साथ गहरा सामंजस्य रखती है, जिसका उपयोग आधुनिक कोडिंग भाषाओं के सिंटैक्स को परिभाषित करने के लिए किया जाता है ।

३.१ चोमस्की पदानुक्रम और पाणिनीय नियमों की शक्ति:

नोम चोमस्की ने भाषाओं को चार स्तरों में विभाजित किया था: नियमित (Type-3), संदर्भ-मुक्त (Type-2), संदर्भ-संवेदनशील (Type-1), और अनप्रतिबंधित (Type-0) । प्रारंभिक विश्लेषणों में पाणिनि के व्याकरण को संदर्भ-मुक्त (Context-free) माना गया था, लेकिन गहराई से अध्ययन

करने पर यह पाया गया कि पाणिनि ने 'संदर्भ-संवेदनशील' (Context-sensitive) नियमों का व्यापक उपयोग किया है। उदाहरण के लिए, संधि के नियम जैसे 'इको यणचि' यह स्पष्ट करते हैं कि वर्ण का परिवर्तन उसके पड़ोसी वर्ण (context) पर निर्भर करता है।

पाणिनि के नियमों की यह संदर्भ-संवेदनशीलता उन्हें अत्यंत शक्तिशाली बनाती है, लेकिन साथ ही वे एक जटिल क्रम (ordering) का पालन करते हैं जिससे वे संगणकीय रूप से सुसाध्य (tractable) बने रहते हैं । यह संतुलन आधुनिक एनएलपी एल्गोरिदम के लिए एक आदर्श प्रस्तुत करता है, जहाँ हम सटीकता और प्रसंस्करण गति के बीच संतुलन बनाने का प्रयास करते हैं ।

३.२ नियम संघर्ष और प्राथमिकता (Conflict Resolution)

जब एक ही स्थिति में दो नियम लागू होते हैं, तो कौन सा नियम प्रभावी होगा? इस 'नियम संघर्ष' (Rule Conflict) को सुलझाने के लिए पाणिनि ने एक अत्यंत परिष्कृत पदानुक्रम विकसित किया । 'विप्रतिषेधे परं कार्यम्' (१.४.२) के अनुसार, समान बल वाले नियमों में अष्टाध्यायी के क्रम में बाद वाला नियम विजयी होता है । इसके अलावा, 'नित्य', 'अन्तरङ्ग', और 'अपवाद' जैसे सिद्धांत यह निर्धारित करते हैं कि कौन सा नियम विशेष है और कौन सा सामान्य। यह तर्क प्रणाली आधुनिक कंपाइलर डिजाइन (Compiler Design) के 'प्रायोरिटी शेड्यूलिंग' और प्रोग्रामिंग भाषाओं में 'ऑपरेटर प्रिसिडेंस' (Operator Precedence) के बिल्कुल समान है । पाणिनि का व्याकरण एक ऐसा एल्गोरिथम है जो कभी 'क्रैश' नहीं होता क्योंकि इसमें हर संभावित टकराव के लिए एक समाधान पहले से ही परिभाषित है ।

४. कारक सिद्धांत और प्राकृतिक भाषा बोध (NLU)

प्राकृतिक भाषा बोध (NLU) का मुख्य लक्ष्य वाक्य के सिंटैक्स (संरचना) से उसका सिमेंटिक्स (अर्थ) निकालना है। पाणिनि का 'कारक' (Karaka) सिद्धांत इस दिशा में सबसे महत्त्वपूर्ण कड़ी है । कारक वे संबंध हैं जो क्रिया और संज्ञा के बीच के अर्थगत संबंध को परिभाषित करते हैं, और वे 'विभक्ति' (Case markers) के माध्यम से व्यक्त होते हैं ।

४.१ कारक और सिमेंटिक भूमिकाएँ [vi]:-

आधुनिक एनएलपी में हम 'सिमेंटिक रोल लेबलिंग' (Semantic Role Labeling) का उपयोग करते हैं ताकि यह पता लगाया जा सके कि वाक्य में 'कर्ता' (Agent), 'विषय' (Theme), और 'साधन' (Instrument) कौन है । पाणिनि ने ६ प्रमुख कारकों को परिभाषित किया है जो इस कार्य को अत्यंत परिशुद्धता के साथ करते हैं :

- कर्ता (Agent): वह जो क्रिया के संपादन में स्वतंत्र है।
- कर्म (Object): वह जिसे कर्ता क्रिया के माध्यम से सबसे अधिक प्राप्त करना चाहता है।
- करण (Instrument): क्रिया की सिद्धि में जो सबसे अधिक सहायक होता है।
- सम्प्रदान (Recipient): जिसके लिए क्रिया की जाती है।
- अपादान (Source): वह निश्चित बिंदु जिससे अलगाव या गति होती है।
- अधिकरण (Locative): क्रिया का आधार या स्थान।

पाणिनीय फ्रेमवर्क यह मानता है कि ये कारक सार्वभौमिक हैं और वक्ता की 'विवक्षा' (इच्छा) पर निर्भर करते हैं । यह अवधारणा आधुनिक एनएलयू प्रणालियों के लिए अत्यंत मूल्यवान है, क्योंकि यह 'फ्री वर्ड ऑर्डर' (मुक्त शब्द क्रम) वाली भाषाओं में अर्थ की स्थिरता बनाए रखने में मदद करती है ।

४.२ आकांक्षा, योग्यता और सन्निधि: अर्थ की खोज : -

एनएलयू में केवल व्याकरणिक शुद्धता पर्याप्त नहीं है; अर्थ का तर्कसंगत होना भी आवश्यक है। भारतीय भाषाविज्ञान की तीन महत्त्वपूर्ण अवधारणाएँ आधुनिक कम्प्यूटेशनल सिमेंटिक्स का आधार बनाती हैं :

- आकांक्षा (Expectancy): एक शब्द की दूसरे शब्द की उपस्थिति की मांग। जैसे 'पीता है' क्रिया को एक कर्म (पानी, दूध आदि) की आकांक्षा होती है ।
- योग्यता (Compatibility): शब्दों के बीच अर्थगत तर्क। 'अग्निना सिञ्चति' (आग से सींचता है) वाक्य व्याकरणिक रूप से सही हो सकता है, लेकिन 'योग्यता' के अभाव में यह अर्थहीन है ।
- सन्निधि (Proximity): शब्दों के बीच की समय और स्थान की निकटता ।

ये तीनों तत्व आधुनिक 'वर्ड सेंस डिसएम्बिग्यूएशन' (WSD) और 'डिपेंडेंसी पार्सिंग' (Dependency Parsing) में महत्त्वपूर्ण भूमिका निभाते हैं । जब कोई मशीन वाक्य का विश्लेषण करती है, तो वह इन कारकों के आधार पर यह निर्णय लेती है कि कौन सा अर्थ सबसे उपयुक्त है।

५. प्राकृतिक भाषा जनन (NLG) और पाणिनीय प्रक्रिया[vii]

ISSN: 2349-6002

प्राकृतिक भाषा जनन (NLG) का कार्य संरचित डेटा को प्राकृतिक भाषा में बदलना है। अष्टाध्यायी की पूरी संरचना एक 'जेनरेटिव इंजन' की तरह है, जहाँ 'व्युत्पत्ति' (Derivation) की प्रक्रिया एक निश्चित पथ का अनुसरण करती है। पॉल किपर्सकी के अनुसार, पाणिनीय प्रणाली में चार स्तर होते हैं जो एनएलजी पाइपलाइन के समान हैं ।

स्तर	पाणिनीय अवधारणा	एनएलजी कार्य
स्तर १	सिमेंटिक/कारक सूचना (Vivaksha)	सामग्री नियोजन और अर्थ निर्धारण।
स्तर २	रूपात्मक रूप (Morphosyntactic)	व्याकरणिक श्रेणियों और प्रत्ययों का चयन।
स्तर ३	अमूर्त रूपात्मक (Abstract Morphological)	प्रत्यय और धातु का संयोजन और परिवर्तन।
स्तर ४	ध्वन्यात्मक आउटपुट (Phonological)	अंतिम वाक्य का निर्माण और ध्वनि अनुकूलन।

५.१ जनन की यंत्रवत प्रकृति :
पाणिनि की 'प्रक्रिया' अत्यंत यंत्रवत है। यदि आपने प्रारंभिक इनपुट (प्रकृति + प्रत्यय) सही चुना है और नियमों का क्रम सही लगाया है, तो परिणाम हमेशा शुद्ध होगा । यह आधुनिक एआई मॉडलों के विपरीत है जो अक्सर 'हैलुसिनेशन' (Hallucination) का शिकार होते हैं। पाणिनीय नियमों को एनएलजी सिस्टम में 'कंस्ट्रेंट्स' (Constraints) के रूप में उपयोग करने से आउटपुट की विश्वसनीयता और शुद्धता बढ़ जाती है ।

५.२ मशीनी अनुवाद [viii]में अनुप्रयोग :-

पाणिनीय व्याकरण का उपयोग मशीनी अनुवाद (Machine Translation) में एक 'इंटरलिंहुआ' (Interlingua) के रूप में किया गया है । चूँकि अधिकांश भारतीय भाषाएं पाणिनि की संरचना के साथ मेल खाती हैं, इसलिए एक भाषा के वाक्य को पहले 'कारक' संरचना में बदलना और फिर उसे दूसरी भाषा में जनरेट करना अत्यंत प्रभावी सिद्ध हुआ है । 'अनुसारक' (Anusaaraka) जैसी प्रणालियाँ इसी सिद्धांत पर आधारित हैं, जो हिंदी से तेलुगु या हिंदी से अन्य भाषाओं के बीच उच्च गति वाला अनुवाद प्रदान करती हैं ।

६. आधुनिक एनएलपी में हाइब्रिड मॉडल और पाणिनीय नियम

वर्तमान में एनएलपी का क्षेत्र 'डीप लर्निंग' (Deep Learning) और 'न्यूरल नेटवर्क्स' (Neural Networks) द्वारा संचालित है। हालाँकि ये मॉडल बहुत अच्छे परिणाम देते हैं, लेकिन इनमें व्याकरणिक सटीकता और तर्क की कमी होती है, विशेष रूप से संस्कृत जैसी 'मोर्फोलॉजिकली रिच' (Morphologically Rich) और 'लो-रिसोर्स' (Low-resource) भाषाओं के लिए । यहाँ पाणिनीय नियमों को न्यूरल मॉडलों के साथ जोड़कर 'हाइब्रिड सिस्टम' विकसित किए जा रहे हैं ।

६.१ नियम-आधारित और डेटा-संचालित मॉडलों का समन्वय :-

हाइब्रिड मॉडलों में, पाणिनीय नियमों का उपयोग 'प्री-प्रोसेसिंग' (Pre-processing) और 'पोस्ट-प्रोसेसिंग' (Post-processing) चरणों में किया जाता है। उदाहरण के लिए, एक 'सैंडी स्प्लिटर' (Sandhi Splitter) जो पाणिनीय नियमों पर आधारित है, एक न्यूरल नेटवर्क को अधिक स्पष्ट डेटा प्रदान कर सकता है, जिससे उसकी सीखने की क्षमता बढ़ जाती है।
शोधों से पता चला है कि संधि-जागरूक (Sandhi-aware) प्री-प्रोसेसिंग से एनएलपी कार्यों की सटीकता में ८% तक की वृद्धि हो सकती है । इसी तरह, 'कारक' सूचना को न्यूरल नेटवर्क के 'अटेंशन मैकेनिज्म' (Attention Mechanism) में एकीकृत करने से मॉडल शब्दों के बीच के संबंधों को अधिक गहराई से समझ सकते हैं ।

६.२ संस्कृत , एनएलपी और डिजिटल संसाधन :-

वर्तमान में कई डिजिटल प्लेटफॉर्म पाणिनीय व्याकरण को आधुनिक सॉफ्टवेयर मॉड्यूल में बदल रहे हैं। इनमें 'संसकृत हेरिटेज प्लेटफॉर्म' और 'डिजिटल कॉर्पस ऑफ संस्कृत' (DCS) प्रमुख हैं । ये संसाधन शोधकर्ताओं को ४,००० सूत्रों को कोड के रूप में उपयोग करने की सुविधा देते हैं, जिससे 'मोर्फोलॉजिकल पार्सिंग' और 'समास डीकम्पोजिशन' जैसे कार्य स्वचालित रूप से किए जा सकते हैं ।

७. ज्ञान निरूपण और एआई: रिक ब्रिग्स की विरासत

१९८५ में रिक ब्रिग्स के लेख ने यह दावा किया था कि संस्कृत के व्याकरणिक ग्रंथ एआई में 'नॉलेज रिप्रेजेंटेशन'

(Knowledge Representation) की समस्या का समाधान प्रदान करते हैं । ब्रिग्स ने तर्क दिया कि संस्कृत की संरचना 'सिमेंटिक नेट्स' (Semantic Nets) के समान है, जहाँ प्रत्येक शब्द का स्थान और अर्थ स्पष्ट रूप से उसके प्रत्यय द्वारा निर्धारित होता है ।

हालांकि आधुनिक एआई 'सिंबॉलिक एआई' (Symbolic AI) से आगे बढ़ चुका है, लेकिन ब्रिग्स का शोध अभी भी 'नॉलेज ग्राफ्स' (Knowledge Graphs) और 'सिमेंटिक वेब' (Semantic Web) प्रौद्योगिकियों के लिए प्रासंगिक है । पाणिनीय व्याकरण यह सिखाता है कि कैसे सूचना को पदानुक्रमित तरीके से व्यवस्थित किया जाए ताकि मशीन उसे बिना किसी भ्रम के पुनः प्राप्त कर सके ।

८. भविष्य की दिशाएँ और निष्कर्ष

पाणिनीय व्याकरण और आधुनिक संगणकीय भाषाविज्ञान का संगम एक नए युग की शुरुआत का संकेत है। जैसे-जैसे हम 'लार्ज लैंग्वेज मॉडल्स' (LLMs) की सीमाओं का अनुभव कर रहे हैं, हमें भाषा के तर्क और संरचना की ओर वापस लौटना पड़ रहा है। पाणिनि की अष्टाध्यायी हमें वह तर्क प्रदान करती है जो मशीनों को वास्तव में 'समझने' (Understand) और 'सृजन' (Generate) करने में मदद कर सकता है ।

भविष्य में, निम्नलिखित क्षेत्र महत्त्वपूर्ण होंगे:

- भारतीय भाषाओं के लिए हाइब्रिड[ix] आर्किटेक्चर: जहाँ पाणिनीय नियमों को ट्रांसफार्मर मॉडल के साथ एकीकृत किया जाएगा ताकि व्याकरणिक रूप से सटीक और संदर्भ-अनुकूल अनुवाद प्राप्त हो सके ।
- स्वचालित व्याकरणिक व्युत्पत्ति: ऐसे शिक्षण उपकरण बनाना जहाँ छात्र किसी भी पद की पाणिनीय व्युत्पत्ति को चरण-दर-चरण देख सकें, जो कोडिंग और भाषा सीखने को जोड़ता हो ।
- लो-रिसोर्स भाषाओं के लिए पाणिनीय मॉडल: संस्कृत की व्याकरणिक गहराई का उपयोग अन्य भारतीय भाषाओं के लिए मॉडल बनाने में करना जिनके पास पर्याप्त डेटा नहीं है ।

अंततः, पाणिनि का कार्य केवल अतीत की धरोहर नहीं है, बल्कि भविष्य की प्रौद्योगिकी का एक सशक्त आधार है। अष्टाध्यायी के सूत्रों में छिपा हुआ गणितीय और तर्कशास्त्रीय सौंदर्य आधुनिक एनएलपी, एनएलयू और एनएलजी की चुनौतियों का समाधान करने की अपार क्षमता रखता है ।

ग्रंथ संदर्भ

इस विषय पर अधिक गहराई से शोध करने के लिए निम्नलिखित प्रमुख ग्रंथों और लेखों का अध्ययन आवश्यक है, जो पाणिनीय व्याकरण और संगणकीय भाषाविज्ञान के सेतु का कार्य करते हैं:

- महर्षि पाणिनि: *अष्टाध्यायी* (व्याकरण का मूल आधार) ।
- भट्टोजि दीक्षित: *सिद्धान्तकौमुदी* (पाणिनीय सूत्रों का प्रक्रियात्मक पुनर्गठन) ।
- पॉल किपर्सकी: *On the Architecture of Panini's Grammar* (संगणकीय संरचना का विश्लेषण) ।
- अक्षर भारती, विनीत चैतन्य और राजीव संगल: *Natural Language Processing: A Paninian Perspective* (भारतीय भाषाओं के लिए आधुनिक एनएलपी अनुप्रयोग) ।
- रिक ब्रिग्स: *Knowledge Representation in Sanskrit and Artificial Intelligence* (एआई मैगजीन, १९८५) ।
- पतंजलि: *व्याकरण महाभाष्य* (व्याकरण के दार्शनिक और तर्कशास्त्रीय आयाम) ।
- जे.ए.एफ. रूडबर्गेन: *Dictionary of Paninian Grammatical Terminology* (पारिभाषिक शब्दावली के लिए संदर्भ) ।
- एस.डी. जोशी और जे.ए.एफ. रूडबर्गेन: *Ashtadhyayi of Panini* (विस्तृत अनुवाद और व्याख्या) ।
- कपिल कपूर: *Dimensions of Panini Grammar* (भारतीय भाषाई परंपरा और आधुनिक विमर्श) ।

ઉલ્લેખિત કાર્યો

[1] Pāṇini grammar is the earliest known computing language, https://doc.gold.ac.uk/aisb50/AISB50-S13/AISB50-S13-Kadvany-paper.pdf

[2] SANSKRIT LANGUAGE AS IN PROGAMMING (NLP) - ijrti, https://www.ijrti.org/papers/IJRTI2504030.pdf

[3] Panini's Structure of Grammar: A Blueprint for Algorithmic Thinking | by Caitanya Desai, https://medium.com/@caitanyadesai/paninis-structure-of-grammar-a-blueprint-for-algorithmic-thinking-80c0f0025277

[4] Fact 94 – There was a school of Sanskrit analysis that was based on semantics (including thoughts on NASA's paper “Knowledge Representation in Sanskrit and Artificial Intelligence”),

https://oursanskrit.com/2020/11/01/fact-94-there-was-a-school-of-sanskrit-analysis-that-was-based-on-semantics-including-thoughts-on-nasas-paper-knowledge-representation-in-sanskrit-and-artificial-intelligence/

[5] Enhancing Panini's Ashtadhyayi: Developing a Comprehensive Text Preprocessing Pipeline Using NLP for Sanskrit Grammar - IJIRT, https://ijirt.org/publishedpaper/IJIRT184431_PAPER.pdf

[6] Identification of Karaka relations in an English sentence - ACL Anthology, https://aclanthology.org/W14-5123.pdf

[7] Pāṇini's grammar and its computerization - Department of Sanskrit Studies, https://sanskrit.uohyd.ac.in/Symposium/papers/houben.pdf

[8] (PDF) Leveraging Pāṇinian Grammar and Neural Models for Morphologically Rich Sanskrit NLP - ResearchGate, https://www.researchgate.net/publication/397068248_Leveraging_Paninian_Grammar_and_Neural_Models_for_Morphologically_Rich_Sanskrit_NLP

[9] (PDF) Hybrid Models in Natural Language Processing - ResearchGate, https://www.researchgate.net/publication/390586059_Hybrid_Models_in_Natural_Language_Processing

[10] NLP vs. NLU vs. NLG: the differences between three natural language processing concepts - IBM, https://www.ibm.com/think/topics/nlp-vs-nlu-vs-nlg

[11] 10 [PDF] Knowledge Representation in Sanskrit and Artificial Intelligence | Semantic Scholar, https://www.semanticscholar.org/paper/Knowledge-Representation-in-Sanskrit-and-Artificial-Briggs/b5258311477908037b500b23fb064e311b140a75

[12] Knowledge Representation in Sanskrit and Artificial Intelligence | AI Magazine, https://ojs.aaai.org/aimagazine/index.php/aimagazine/article/view/466

[i] 1. Pāṇini grammar is the earliest known computing language, https://doc.gold.ac.uk/aisb50/AISB50-S13/AISB50-S13-Kadvany-paper.pdf

[ii] 2. SANSKRIT LANGUAGE AS IN PROGAMMING (NLP) - ijrti, https://www.ijrti.org/papers/IJRTI2504030.pdf

[iii] Panini's Structure of Grammar: A Blueprint for Algorithmic Thinking | by Caitanya Desai, https://medium.com/@caitanyadesai/paninis-structure-of-grammar-a-blueprint-for-algorithmic-thinking-80c0f0025277

[iv] Fact 94 – There was a school of Sanskrit analysis that was based on semantics (including thoughts on NASA's paper "Knowledge Representation in Sanskrit and Artificial Intelligence"), https://oursanskrit.com/2020/11/01/fact-94-there-was-a-school-of-sanskrit-analysis-that-was-based-on-semantics-including-thoughts-on-nasas-paper-knowledge-representation-in-sanskrit-and-artificial-intelligence/

[v] Enhancing Panini's Ashtadhyayi: Developing a Comprehensive Text Preprocessing Pipeline Using NLP for Sanskrit Grammar - IJIRT, https://ijirt.org/publishedpaper/IJIRT184431_PAPER.pdf

[vi] Identification of Karaka relations in an English sentence - ACL Anthology, https://aclanthology.org/W14-5123.pdf

[vii] Pāṇini's grammar and its computerization - Department of Sanskrit Studies, https://sanskrit.uohyd.ac.in/Symposium/papers/houben.pdf

[viii] (PDF) Leveraging Pāṇinian Grammar and Neural Models for Morphologically Rich Sanskrit NLP - ResearchGate, https://www.researchgate.net/publication/397068248_Leveraging_Paninian_Grammar_and_Neural_Models_for_Morphologically_Rich_Sanskrit_NLP

[ix] (PDF) Hybrid Models in Natural Language Processing - ResearchGate, https://www.researchgate.net/publication/390586059_Hybrid_Models_in_Natural_Language_Processing