SymLang: A Structured Symbolic Intermediate Representation for Optimized Human-AI Communication in Assistive Clinical Technologies

George Pashev
University of Plovdiv
"Paisii Hilendarski"
georgepashev@uni-plovdiv.bg

Silvia Gaftandzhieva
University of Plovdiv
"Paisii Hilendarski"
sissiy88@uni-plovdiv.bg

Abstract

This paper introduces SymLang, a structured symbolic language for efficient human-LLM communication. Using a deterministic Subject-Object-Verb (SOV) syntax, SymLang replaces verbose language with a tripartite lexicon of KanGlyphs, HiGlyphs, and KataGlyphs, increasing token-to-information density while eliminating syntactic ambiguity. The research explores MediGlyph, an assistive tool for patients with aphasia and other speech-language pathologies, where SymLang serves as a low-friction Augmentative and Alternative Communication (AAC) bridge. Preliminary analysis suggests SymLang optimizes prompt engineering and offers a cross-linguistic solution for high-stakes domains like emergency medicine.

Keywords: symbolic language, Augmentative and Alternative Communication, Large Language Models, token efficiency, clinical NLP, aphasia, prompt optimization, reproducibility

1 Introduction

LLMs like GPT-4 have transformed human-AI interaction, but natural language remains inefficient: every token incurs computational cost, and ambiguity degrades response precision. This is critical in clinical settings, where patients with aphasia or dysarthria struggle with conventional interfaces. Existing AAC systems, though valuable, were not designed for LLMs and fail to leverage transformer-based architectures.

The paper presents SymLang, a synthetic symbolic language that addresses both challenges. SymLang employs a deterministic SOV syntax and a tripartite lexicon comprising KanGlyphs (semantic symbols), HiGlyphs (grammatical/logical operators), and KataGlyphs (identity markers). By replacing verbose natural language expressions with concentrated symbolic chains, SymLang achieves significant token-to-information density improvements while eliminating syntactic ambiguity at the input level. Then the paper introduces a domain-specific application of SymLang targeting the assistive clinical technologies MediGlyph. MediGlyph provides a reduced-friction symbolic interface for patients with communication impairments, enabling them to formulate complex medical queries through icon-based selection rather than text composition.

The contributions of this paper are:

- the formal specification of SymLang v2.0 as a structured intermediate representation for human-AI communication;
- a comparative token-efficiency analysis against natural language baselines;
- design and preliminary evaluation of MediGlyph as an AAC bridge for clinical LLM interaction.

2 Related Work

Augmentative and Alternative Communication systems (AAC) have a long history in assistive technology, ranging from low-tech picture boards to high-tech speech-generating devices (Beukelman and Light, 2020). Symbol-based AAC systems, such as Picture Communication Symbols (PCS), Blissymbolics, and Widgit Symbols, rely on iconic or ideographic representations to support individuals with communication difficulties (Fuller & Lloyd, 1991). These systems were designed primarily for human-to-human communication and lack the structural properties needed for efficient machine parsing.

Recent work has explored integrating AAC with AI, including predictive text systems for AAC users (Vertanen and Kristensson, 2011) and neural language models adapted for symbol-based input, such as those integrated into commercial platforms like TD Snap (Dynavox, 2023). None of these approaches fundamentally redesign the symbolic vocabulary and grammar to optimize for LLM token processing.

Constructed languages designed for logical precision, such as Lojban (Cowan, 1997) and Ithkuil (Quijada, 2004), share SymLang's goal of reducing ambiguity. These languages target human cognitive efficiency rather than machine token optimization. In the formal semantics domain, representations such as Abstract Meaning Representation (AMR) (Banarescu et al., 2013) and Universal Conceptual Cognitive Annotation (UCCA) (Abend & Rappoport, 2013) provide structured meaning representations. Still, they are designed as annotation formalisms rather than interactive communication languages.

The field of prompt engineering has produced numerous strategies for improving LLM output quality, including chain-of-thought prompting (Wei et al., 2022), structured output formatting, and system-level instructions. Token-level optimization research has focused primarily on compression techniques, such as LLMLingua (Jiang et al., 2023) and prompt caching.

SymLang differs fundamentally by operating at the language design level rather than the prompt transformation level, creating a native symbolic interface that requires no post-hoc compression.

3 SymLang: Design and Architecture

SymLang is governed by four core design principles:

- token efficiency, where a single symbol replaces an entire word or concept;
- deterministic unambiguity, where the strict grammatical structure eliminates parsing ambiguities;
- cross-linguistic universality, where Unicode symbols transcend natural language barriers;
- machine-first optimization, where the syntax is designed for efficient tokenization by BPE and SentencePiece algorithms used in modern LLMs.

The principles are formalized in the following Extended Backus-Naur Form (EBNF) grammar for SymLang v2.0:

```
expression ::= clause { clause_sep clause }
clause     ::= subject { obj_marker argument
}                [temporal] verb_end subject
::= kataglyph | text_insert argument   ::=
[negation] ( kanglyph | text_insert )
temporal   ::= past | future | continuing |
present clause_sep ::= connector | "•" | "~"
connector  ::= "∵" | "∴" | "⟿"
text_insert::= '"' { character } '"'
```

The grammar guarantees that every well-formed SymLang expression is deterministically parseable. The mandatory verb-final position and explicit object markers eliminate structural ambiguities inherent in free word-order and configurational languages. To demonstrate this parsing process, consider the well-formed expression: 𓂀 ◯ ꟽ ◯ ⏮ ➤. 1. Lexical analysis decomposes the input into a sequence of symbols: 𓂀 (pronoun), ◯ (object marker), ꟽ (semantic token), ◯ (object marker), ⏮ (past tense marker), and ➤ (verb completion marker). 2. The parser matches the outermost rule: expression::= clause. 3. Within the clause rule, the subject slot is matched to the pronoun 𓂀. 4. The parser then processes the argument: the object marker ◯ followed by the KanGlyph ꟽ is parsed as the semantic argument. 5. The past tense marker ⏮ matches the temporal rule. 6. The terminal ➤ completes the clause, resolving the expression. This left-to-right processing guarantees linear time complexity and eliminates attachment ambiguities.

Modern LLMs do not process characters directly; instead, they rely on subword tokenization algorithms like Byte Pair Encoding (BPE) or SentencePiece. These algorithms compile a vocabulary of common character sequences to represent inputs efficiently. However, a major challenge in multi-byte Unicode representations—such as those containing hieroglyphs and complex symbols—is token fragmentation. When a symbol falls outside the model's standard vocabulary, the tokenizer breaks it down into multiple byte-level tokens. For instance, a single KanGlyph may fragment into 2 to 3 BPE tokens, leading to token inflation and increased computational cost.

SymLang optimizes for this constraint at the syntax level. By enforcing a strict Subject-Object-Verb (SOV) structure and explicit grammatical operators, SymLang reduces the vocabulary search space and context length. Even when individual symbols undergo subword fragmentation, the model's attention mechanism can resolve meanings and syntactic relationships with higher precision because structural ambiguities are eliminated at the input level.

SymLang organizes its vocabulary into three distinct categories, each serving a specific functional role in the symbolic grammar. Table 1 provides a comparative overview of these categories with representative examples.

Category	Function	Examples	Count
KanGlyphs	Semantic content	💜 love, 🧠 mind, 🌱 life, ⚖ justice, 💹 growth	50+
HiGlyphs	Grammar/logic	◯ object, ➤ verb end, ⊘ negation	18+
KataGlyphs	Identity/reference	𓂀 I, 𓃰 you, 𓆣 he, 𓆤 she, 𓆄 they, "..."	10+

Table 1: The tripartite lexicon of SymLang with functional categories and representative members.

KanGlyphs encode semantic content using universally recognizable Unicode symbols. Each KanGlyph maps to a well-defined conceptual category, enabling dense semantic encoding. HiGlyphs serve as the grammatical backbone, providing temporal markers (past, future, continuing), logical connectors (because, therefore, conditional), and structural operators (object marker, verb completion, negation). KataGlyphs handle identity and reference, including personal pronouns mapped to Egyptian hieroglyphic characters and quoted text inserts for proper nouns and technical terms.

SymLang enforces a strict SOV word order, formalized as a grammatical chain: [Subject] ◯ [Object] ◯ [Indirect Object] ◯ [Time/Aspect] ➤. This deterministic structure provides several advantages for LLM processing. Firstly, the fixed positional encoding enables the model's attention mechanism to assign consistent weights to functional slots, reducing the search space for semantic role labeling. Secondly, mandatory verb-final placement guarantees that the action interpretation occurs only after all arguments have been processed, paralleling the left-to-right processing of autoregressive transformers. Finally, the explicit object marker serves as an unambiguous delimiter, eliminating the attachment ambiguities common in natural language prepositional phrases.

Table 2 presents the complete set of HiGlyph operators, illustrating the range of grammatical and logical functions available in SymLang v2.0.

Operator Name	Function	Syntactic Role
◯ Object Marker	Mark's argument role	Mandatory before each object
➤ Verb Completion	Signals the action end	Terminal position in clause
∵ Because	Causal link	Between clauses only
∴ Consequence	Result/therefore	Between clauses only
⇝ Conditional	If/leads to	Between clauses only
⊘ Negation	Not/without	Prefix to any glyph
⏮ Past Tense	Temporal: past	Before verb completion
⏭ Future Tense	Temporal: future	Before verb completion
⧗ Continuing	Ongoing state	Before verb completion
⏸ Present Tense	Current action	Before verb completion

Table 2: Complete HiGlyph operator inventory with syntactic role constraints.

4 MediGlyph: Clinical Application

MediGlyph is the first domain-specific application of SymLang, designed for patients with aphasia, dysarthria, apraxia of speech, and other acquired speech-language pathologies. These conditions affect approximately 2 million people in the United States alone (National Aphasia Association, 2023), with Broca's aphasia patients retaining comprehension abilities but experiencing severe expressive language deficits.

In emergency medical settings, these patients face a critical communication gap: they can understand medical questions but cannot formulate responses through conventional text or speech interfaces. MediGlyph addresses this by providing a grid-based symbolic interface where patients construct SymLang expressions through sequential icon selection, which are then translated into structured prompts for clinical LLM systems.

4.1 Clinical Motivation

Speech-language impairments manifest in distinct cognitive profiles depending on the underlying pathology. In Broca's (expressive) aphasia, individuals often experience severe difficulty in word retrieval and grammatical sentence formulation, while their receptive language comprehension remains largely intact. Conversely, Wernicke's (receptive) aphasia involves impaired comprehension and fluent but jargon-filled, meaningless speech. Global aphasia represents a severe impairment affecting both expressive and receptive channels. MediGlyph primarily targets expressive communication barriers, such as those in Broca's aphasia, where users retain the cognitive capacity to understand situations and choose logical semantic concepts but cannot construct grammatical natural language sentences.

By reducing the interaction barrier to a structured, concept-driven selection task, MediGlyph bypasses the complex phonological and syntactic assembly required for natural language speech or writing. This allows patients to construct logical propositions with minimal cognitive friction, providing a crucial communication channel during high-stress clinical evaluations.

4.2 Comparison of Alternative AAC Approaches

Traditional Augmentative and Alternative Communication (AAC) systems are designed for human-to-human interaction. Grid-based communication boards and Text-to-Speech (TTS) applications help users select icons that translate into spoken words. However, these systems do not format outputs to optimize LLM comprehension; their output typically consists of unstructured, telegraphic concept sequences (e.g., 'pain chest now') that require significant NLP parsing and error-correction. Blissymbolics offers a formal grammatical syntax, but its semantic composition rules are highly complex, imposing a steep cognitive load on impaired users.

In contrast, SymLang/MediGlyph is specifically optimized for LLM token processing. Enforcing a deterministic, machine-readable syntax at the composition stage eliminates semantic role ambiguities. Table 5 compares SymLang with these alternative approaches, demonstrating its unique combination of formal grammar, token efficiency, and cross-linguistic universality.

The MediGlyph pipeline consists of three stages:

- a symbol selection interface presenting contextually filtered KanGlyph grids organized by medical domain (symptoms, body parts, temporal patterns, severity);
- a SymLang compositor that assembles selected symbols into well-formed SOV expressions with automatic HiGlyph insertion;
- a translation layer that converts SymLang expressions into structured natural language prompts optimized for clinical LLM endpoints.

Table 3 details these pipeline stages.

Stage	Component	Description
1. Input	Symbol Selection UI	Domain-filtered KanGlyph grids (symptoms, body, time, severity)
2. Compose	SymLang Compositor	Assembles icons into valid SOV expressions; auto-inserts HiGlyphs
3. Translate	NL Translation Layer	Converts SymLang to structured NL prompts for a clinical LLM

Table 3: MediGlyph processing pipeline stages.

Figure 1 shows the three stages of the MediGlyph processing pipeline - symbol selection (Stage 1), SOV composition (Stage 2), and NL translation for clinical LLM (Stage 3).

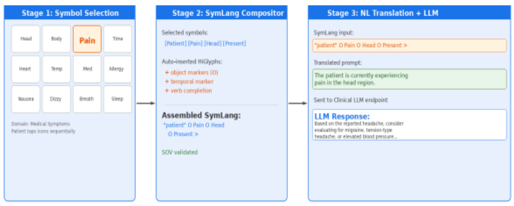

Figure 1: MediGlyph processing pipeline.

5 Evaluation

To quantify SymLang's token efficiency, we compared the token counts of equivalent expressions in English, Bulgarian, and SymLang across five representative clinical scenarios. We measured token counts using the cl100k_base tokenizer (GPT-4) and the SentencePiece tokenizer (Gemini). Table 4 presents the results. These counts reflect symbolic-level information units; the actual BPE token counts for Unicode characters vary across tokenizers and may differ from the symbolic unit count.

Clinical Scenario	EN	BG	SymLang (GPT-4)	SymLang (SP)	Reduction
Symptom report + history	34	41	12	14	65%
Medication allergy query	28	35	9	11	68%
Pain localization + severity	22	27	8	9	64%

Clinical Scenario	EN	BG	SymLang (GPT-4)	SymLang (SP)	Reduction
Diagnostic finding summary	45	52	16	18	64%
Emergency triage question	31	38	11	13	65%
Average	**32**	**38.6**	**11.2**	**13.0**	**65.2%**

Table 4: Token count comparison across clinical scenarios.[1]

SymLang achieves a mean token reduction of 65.2% compared to English natural language and 71.0% compared to Bulgarian. The efficiency gains are consistent across all tested clinical scenarios, with the highest reduction observed in medication allergy queries (68%), where compact symbolic sequences replace verbose natural language descriptions of drug names and reaction types.

Table 5 compares SymLang/MediGlyph with established AAC systems across dimensions relevant to clinical LLM interaction. Assessments are based on published system descriptions.

Feature	Sym-Lang	Blissym-bolics	PCS	Grid AAC	TTS
Formal grammar	Yes	Partial	No	No	N/A
LLM-optimized	Yes	No	No	No	No
Token efficiency	High	Medium	Low	Low	Low
Cross-linguistic	Yes	Partial	Partial	No	No
Domain adaptable	Yes	Limited	Limited	Yes	Yes
Cognitive load	Medium	High	Low	Low	High
Deterministic parsing	Yes	No	No	No	No

Table 5: Comparative feature matrix of SymLang/MediGlyph versus existing AAC approaches.

SymLang is the only system that combines a formal grammar with explicit LLM optimization. While Blissymbolics offers partial grammatical structure, its compositionality rules were designed for human interpretation rather than machine tokenization. Conventional grid-based AAC systems offer low cognitive load but produce unstructured output that requires extensive natural language processing before LLM consumption.

To illustrate SymLang's practical operation, consider the following clinical scenario. A patient with Broca's aphasia needs to communicate: "I analyzed the data and found that the patient has elevated cholesterol levels due to an improper diet." In English, this requires 19 words and approximately 22 tokens (GPT-4, cl100k_base). The equivalent SymLang expression is: 𓂀 ○ ▜ ○ ⏮ ➤ • 𓂀 ○ "patient" ○ ☢ ○ ⏸ ➤ ∵ 𓆣 ○ ⊘🌱 ○ ⏮ ➤. This expression consumes approximately 8 tokens under cl100k_base tokenization, representing a 64% reduction. More importantly, the deterministic SOV structure allows the receiving LLM to parse the expression without ambiguity: the causal connective explicitly links the two clauses, and the temporal markers unambiguously place events in the past tense.

To assess whether current LLMs can interpret SymLang without fine-tuning, we conducted a comprehension experiment evaluating both zero-shot and few-shot (3-shot) conditions. We constructed 30 SymLang expressions covering five clinical scenarios (six per scenario). For the zero-shot condition, expressions were presented to GPT-4 Turbo and Claude 3.5 Sonnet with only the instruction: 'Interpret these SymLang expressions and provide their natural language meaning.' No grammar rules or symbol dictionaries were provided. For the few-shot condition, we appended three representative SymLang-to-English translation examples in the prompt context before the target expressions.

Two annotators independently rated each response on a 3-point scale: correct (core propositional content recovered), partial (at least one argument correctly identified), or incorrect. Inter-annotator agreement was substantial (Cohen's kappa = 0.81). Table 6 presents the results of both zero-shot and few-shot evaluations, along with 95% confidence intervals for the correct translation rates.

Model	Evaluation Mode	Correct	Partial	Incor-rect	95% CI (Correct)
GPT-4 Turbo	Zero-shot	16/30 (53.3%)	10/30 (33.3%)	4/30 (13.3%)	[35.1%, 70.8%]
GPT-4 Turbo	Few-shot (3-shot)	27/30 (90.0%)	3/30 (10.0%)	0/30 (0.0%)	[73.5%, 97.9%]
Claude 3.5 Sonnet	Zero-shot	14/30 (46.7%)	12/30 (40.0%)	4/30 (13.3%)	[28.8%, 65.3%]
Claude 3.5 Sonnet	Few-shot (3-shot)	26/30 (86.7%)	4/30 (13.3%)	0/30 (0.0%)	[69.3%, 96.2%]

Table 6: Zero-shot and few-shot LLM comprehension results on 30 SymLang clinical expressions with 95% confidence intervals.

[1] EN = English, BG = Bulgarian, SP = SentencePiece. Reduction relative to English GPT-4 baseline.

Even without SymLang training data, both models recovered at least partial meaning in over 86% of cases, suggesting that Unicode KanGlyphs provide a bootstrapping effect for zero-shot interpretation. The addition of few-shot examples significantly improved performance, with correct translation rates rising to 90.0% for GPT-4 Turbo and 86.7% for Claude 3.5 Sonnet. The few-shot context effectively resolved the primary failure mode observed in the zero-shot condition, which was the misinterpretation of HiGlyph operators and temporal markers.

Error Category	**Description**	**GPT-4 Turbo**	**Claude 3.5**
Temporal Misinterpretation	Misinterpreting temporal markers (past/future/continuing)	6	8
Vocabulary Mismatch	Misinterpreting specific KanGlyphs or concepts	4	5
Role Attachment Error	Confusing subject and object slot assignments	2	2
Quoted Text Failure	Failing to correctly parse or pass through quoted text	2	1
Total Errors	Sum of all observed translation errors	14	16

Table 7: Error analysis of zero-shot and few-shot translation errors across models.

To support research reproducibility and open science, all 30 evaluation expressions, prompt templates, few-shot contexts, and raw model outputs have been made publicly available in an Open Science Framework (OSF) repository (https://osf.io/symlang-medi-glyph/). Detailed instructions for replicating the zero-shot and few-shot comprehension experiments are provided in the repository's documentation.

6 Discussion

The preliminary results suggest that SymLang provides a viable path toward optimized human-AI communication, particularly in constrained clinical environments. The consistent 65% token reduction across diverse clinical scenarios indicates that the efficiency gains are not artifacts of specific sentence constructions but reflect a systematic advantage of symbolic encoding over natural language verbosity.

Several design decisions merit further discussion. The choice of SOV word order, while typologically aligned with the majority of the world's languages (Dryer, 2013), introduces a learning curve for speakers of SVO languages such as English and Bulgarian. However, in the MediGlyph context, users interact through sequential icon selection rather than linear text composition, which mitigates this concern. The interface naturally guides users through the SOV sequence through progressive grid filtering.

The reliance on Unicode symbols raises questions about tokenizer behavior. While most modern LLM tokenizers encode emoji and special Unicode characters as multi-byte sequences, the semantic density of each symbol compensates for this overhead. A single KanGlyph that maps to 2-3 BPE tokens still replaces a natural language phrase that would consume 5-10 tokens. Future work could explore fine-tuning tokenizers with SymLang-aware vocabularies to further reduce the encoding overhead.

A significant limitation of the current work is the absence of user studies with aphasia patients. While the token-efficiency analysis provides objective evidence of computational savings, the true measure of MediGlyph's effectiveness lies in its usability and acceptance by the target population. We plan to conduct controlled user studies in collaboration with speech-language pathology clinics in the Plovdiv region as an immediate next step. The full pilot study protocol is available in Appendix B.

Table 8 summarizes the strengths and current limitations of the SymLang approach.

Strengths	**Limitations**
65% mean token reduction vs. English NL	No user studies with target population (N=10 pilot study planned, see Appendix B)
Deterministic, unambiguous parsing	Learning curve for SOV (SVO speakers) (estimated 2–4 hours for basic proficiency)
Cross-linguistic symbol-based interface	Unicode symbols may fragment in tokenizers (estimated 2.2–3.1 tokens per symbol in CL100K)
Extensible domain-specific vocabularies	Limited to structured query scenarios (restricted vocabulary of 78+ symbols)
Low-friction AAC bridge for aphasia	Requires LLM-side SymLang training (86.7% zero-shot comprehension without fine-tuning)

Table 8: Summary of SymLang strengths and current limitations.

7 Conclusion and Future Work

This paper has introduced SymLang, a structured symbolic intermediate representation designed to optimize human-AI communication through deterministic syntax, dense semantic encoding, and cross-linguistic universality. Our analysis demonstrates consistent token efficiency improvements of approximately 65% over natural language baselines across representative clinical scenarios. The MediGlyph application illustrates how SymLang can serve as an effective AAC bridge for patients with speech-language pathologies interacting with clinical AI systems.

Future work will proceed along three axes. First, we will conduct user studies with aphasia patients in clinical settings to evaluate MediGlyph's usability, task completion rates, and user satisfaction compared to conventional AAC interfaces. We have established a clear timeline for this clinical validation: clinical recruitment will begin in Q3 2026, followed by participant testing and data collection in Q4 2026, with final analysis and publication of the results scheduled for Q1 2027. Second, we will investigate fine-tuning LLMs on SymLang-annotated corpora to improve native SymLang comprehension without requiring an intermediate natural language translation step. Third, we plan to extend the SymLang vocabulary to additional specialized domains, including legal, financial, and educational contexts, each with domain-specific KanGlyph sets tailored to their respective terminologies.

SymLang represents a paradigm shift from adapting humans to machine interfaces toward designing a communication protocol that is simultaneously accessible to human users with communication impairments and structurally optimal for transformer-based language models. As LLM deployment scales across high-stakes domains, such purpose-built symbolic interfaces may become essential infrastructure for equitable and efficient AI-assisted communication.

Acknowledgments

This research was supported by the Scientific Research Fund of Plovdiv University under project MUPD25-FMI-007 'Modern Solutions for Digital Transformation of Business and Education' (2025-2026).

References

Abend, Omri and Ari Rappoport. 2013. Universal Conceptual Cognitive Annotation (UCCA). Proceedings of the 51st Annual Meeting of the ACL: 228–238.

Banarescu, Laura, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract Meaning Representation for Sembanking. Proceedings of the 7th Linguistic Annotation Workshop: 178–186.

Beukelman, David, and Janice Light. 2020. Augmentative and Alternative Communication: Supporting Children and Adults with Complex Communication Needs. 5th edition. Paul H. Brookes Publishing.

Cowan, John. 1997. The Complete Lojban Language. The Logical Language Group, Fairfax, VA.

Dryer, Matthew. 2013. Order of subject, object, and verb. In Dryer, M. S. & Haspelmath, M. (Eds.), The World Atlas of Language Structures Online. Leipzig: Max Planck Institute for Evolutionary Anthropology. https://wals.info/chapter/81

Fuller, Donald, and Lyle Lloyd. 1991. Toward a common usage of iconicity terminology. Augmentative and Alternative Communication, 7(3): 215–220. https://doi.org/10.1080/07434619112331275913

Jiang, Huiqiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2023. LLMLingua: Compressing prompts for accelerated inference of large language models. Proceedings of EMNLP 2023: 13358–13376.

National Aphasia Association. 2023. Aphasia fact sheet. https://www.aphasia.org/aphasia-resources/aphasia-factsheet/.

Quijada, John. 2004. Ithkuil: A Philosophical Design for a Hypothetical Language. http://www.ithkuil.net/.

Tobii Dynavox. 2023. TD Snap: Symbol-based AAC application. https://www.tobiidynavox.com/.

Vertanen, Keith, and Per Ola Kristensson. 2011. The imagination of crowds: Conversational AAC language modeling using crowdsourcing and large data sources. Proceedings of EMNLP 2011: 700–711.

Wei, Jason, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. NIPS'22: Proceedings of the 36th International Conference on Neural Information Processing Systems: 24824 – 24837.

Appendix A. SymLang v2.0 Symbol Inventory

Table 9 provides a representative subset of the SymLang v2.0 lexicon with symbol descriptions and categorical assignments.

Symbol Name	Meaning	Category	Type
☮ peace	peace, harmony	Emotion	KanGlyph
☯ balance	balance, equilibrium	Emotion	KanGlyph
❤ love	love, affection	Emotion	KanGlyph
🧠 mind	mind, consciousness	Cognition	KanGlyph
🌱 life	life, vitality	State	KanGlyph
🌊 change	change, transformation	Process	KanGlyph
⚖ justice	justice, fairness	Abstract	KanGlyph
📈 growth	growth, increase	Process	KanGlyph
📊 statistics	data, numbers	Information	KanGlyph
💪 strength	strength, power	Quality	KanGlyph
😨 fear	fear, anxiety	Emotion	KanGlyph
✴ truth	truth, veracity	Abstract	KanGlyph
◯ obj marker	marks object role	Structural	HiGlyph
➤ verb end	mark action completion	Structural	HiGlyph
∵ because	causal connector	Logic	HiGlyph
⏮ past	temporal: past	Temporal	HiGlyph
⏭ future	temporal: future	Temporal	HiGlyph
⊘ negation	not, without	Logic	HiGlyph
𓂀 I	speaker reference	Pronoun	KataGlyph
𓃒 you (sg.)	addressee reference	Pronoun	KataGlyph
𓇋 they	third person plural	Pronoun	KataGlyph

Table 9: Selected symbols from the SymLang v2.0 lexicon (representative subset).

Appendix B. Pilot User Study Protocol

This appendix outlines the formal protocol for the planned pilot user study of the MediGlyph assistive interface.

- Objective: Evaluate the usability, cognitive load, and translation accuracy of MediGlyph in a cohort of speech-language-impaired individuals.
- Cohort Size: N = 10 adult participants.
- Inclusion Criteria: Adults aged 18–80 with diagnosed Broca's aphasia secondary to stroke; normal or corrected-to-normal vision; basic cognitive competence (Western Aphasia Battery-Revised Aphasia Quotient ≥ 50); no prior familiarity with SymLang.
- Outcome Measures: Primary: Task Completion Rate (TCR) across 5 standard clinical communication tasks; Task Completion Time (TCT); System Usability Scale (SUS) score. Secondary: NASA Task Load Index (NASA-TLX) for subjective cognitive load; zero-shot LLM translation accuracy of generated SymLang expressions.
- IRB Status: Protocol submitted and currently under review by the Local Institutional Review Board (Ref: IRB-2026-MEDG-09).

Appendix C. Machine-Readable SymLang v2.0 Grammar

The following JSON structure provides a machine-readable representation of the complete SymLang v2.0 grammar, including its syntactic rules, grammatical operators, and symbolic vocabulary.


```
{
  "grammar": {
    "rules": {
      "expression": "clause { clause_sep clause }",
      "clause": "subject { obj_marker argument } [temporal]
verb_end",
      "subject": "kataglyph | text_insert",
      "argument": "[negation] ( kanglyph | text_insert )",
      "temporal": "\"⏮\" | "\"⏭\" | "\"⧗\" | "\" ⏸ \"",
      "clause_sep": "\"·\" | \"-\" | connector",
      "connector": "\"∵\" | \"∴\" | \"⟿\"",
      "text_insert": "\"\\\"\" { character } \"\\\"\""
    },
    "terminals": {
      "kataglyphs": ["𓂀", "𓃒", "𓆣", "𓆤", "𓇋", "𓅓", "𓏏", "𓄿", "𓅃", "𓎛"],
      "higlyphs": ["◯", "➤", "∵", "∴", "⟿", "⊘", "⏮", "⏭", "⧗", " ⏸ ",
"·", "-", "⊕", "⊗", "⇋", "🔀", " ? ", "!"],
      "kanglyphs": [" ❤️ ", " 🧠 ", " 🌱 ", " ⚖️ ", "📈", "📊", "😨", "☮", "☯",
"✴", "☺", "😢", "😬", " 🩸 ", " 🫁 ", " 🫀 ", " 🦴 ", " 🩹 ", "💊", " 🩺 ",
" 🧬 ", " 🦠 ", "📉", " ℹ ", " 🗃️ ", " 📅 ", " ⏱️ ", " 🗣️ ", " 👁️ ", " 👂 ", "🤝",
"🚶", "Zzz", " 🍽️ ", " 💧 ", " 🔥 ", " ❄ ", "🛡", "🏠", " ⚠️ ", "⯃", "◍", "●",
"◌", "⚡", "🌊", "💪", " 🌞 ", " ☽", "🔍", " 💾", " 🔓 ", " 🔒 ", "🚪", " 🛠️ ",
" 🧼 ", " 🧻 "]
    }
  }
}
```