International Journal of Advance and Innovative Research
Volume 13, Issue 2: April - June 2026
ISSN 2394 - 7780

AI BASED INTELLIGENT WORD GENERATION SYSTEM FOR ISL USING DICTIONARY CONSTRAINED ANAGRAM AND PREFIX BASED SEARCH

Swastik Manglam[a], Dr. Poonam Dahiya[b] and Dr. Gaurav Aggarwal[c]
[a]Research Scholar, Department of CSE, Jagannath University, Bahadurgarh, Jhajjar (Haryana)
[b]Associate Professor, Department of CSE, Jagannath University, Bahadurgarh, Jhajjar (Haryana)
[c]Professor, Department of CSE, Jagannath University, Bahadurgarh, Jhajjar (Haryana)

ABSTRACT
Sign language translation systems often rely on recognizing individual gestures but face challenges when converting sequences of recognized alphabets into meaningful words. This research presents an ***AI-based intelligent word generation*** *system designed to transform sequences of recognized ISL alphabets into valid and meaningful words. The proposed system employs a dictionary-constrained generation approach combined with prefix-based pruning to efficiently generate candidate words from unordered or partially ordered input sequences.*

Unlike traditional brute-force permutation methods that suffer from high computational complexity, the proposed method utilizes a prefix-guided search strategy to eliminate invalid combinations early in the generation process. The system integrates a structured dictionary representation using a Trie, enabling efficient validation of prefixes and complete words. Additionally, a ranking mechanism is introduced to prioritize commonly used words, and a human-in-the-loop selection module ensures accurate disambiguation.

Experimental evaluation indicates that the system significantly reduces computational overhead while maintaining high accuracy in word generation. This work serves as a crucial intermediate layer between gesture recognition and sentence-level translation, contributing to the development of a complete and practical ISL translation system.

INTRODUCTION
Indian Sign Language (ISL) translation systems primarily focus on recognizing individual gestures corresponding to alphabets or numbers. While such systems form the foundation of sign language interpretation, a major challenge lies in converting sequences of recognized alphabets into meaningful words. Direct concatenation of recognized alphabets often results in ambiguous or incorrect outputs, especially when gesture recognition is imperfect or when the order of detected alphabets is inconsistent.

In real-world scenarios, users may produce gestures at varying speeds, with slight delays or recognition errors, leading to unordered or partially incorrect sequences. Traditional approaches attempt to solve this problem by generating all possible permutations of input alphabets and matching them against a dictionary. However, such brute-force methods are computationally expensive, with factorial time complexity, making them impractical for real-time applications.

To address this issue, the proposed system introduces a dictionary-constrained word generation mechanism that builds candidate words incrementally using prefix validation. Instead of generating all permutations, the system explores only those combinations that correspond to valid dictionary prefixes, thereby significantly reducing the search space.

Furthermore, ambiguity in generated words is handled through a human-in-the-loop interaction model, where multiple candidate words are presented to the user for selection. This ensures accuracy while maintaining system efficiency. The proposed approach bridges the gap between low-level gesture recognition and higher-level language understanding, forming an essential component of a complete ISL translation framework.

PROBLEM STATEMENT
Although gesture recognition systems can accurately identify individual alphabets, converting these outputs into meaningful words remains a complex challenge. The primary difficulty arises from the unordered or partially ordered nature of recognized alphabet sequences, which may result from recognition errors or variations in gesture input.

Existing approaches that rely on generating all possible permutations of input alphabets are computationally inefficient and unsuitable for real-time applications. Additionally, these methods often produce multiple ambiguous outputs without an effective mechanism for selecting the correct word.

ISSN 2394 - 7780

Another key limitation is the absence of intelligent filtering during word generation, leading to unnecessary computation on invalid combinations. This not only increases processing time but also reduces system scalability.

Therefore, there is a need for an efficient and intelligent word generation system that can convert sequences of alphabets into valid words while minimizing computational complexity and handling ambiguity effectively.

RESEARCH GAP

Despite progress in gesture recognition, the following gaps exist in word generation systems:

1. High Computational Complexity

Most existing methods rely on brute-force permutation generation, resulting in factorial time complexity and poor scalability.

2. Lack of Dictionary-Constrained Search

There is limited use of structured dictionary-based approaches that can guide the generation process efficiently.

3. Absence of Prefix-Based Pruning

Existing systems do not effectively eliminate invalid combinations early, leading to unnecessary computations.

4. Ambiguity in Word Selection

Multiple valid words may be generated from the same set of alphabets, but most systems lack mechanisms for resolving ambiguity.

5. Limited Integration with Recognition Systems

There is insufficient integration between gesture recognition outputs and higher-level word generation modules.

Resolving these challenges is essential for developing a reliable and efficient ISL translation system that delivers accurate results while remaining accessible and user-friendly, particularly for children and other users with special communication needs.

RESEARCH OBJECTIVES

The primary objective of this research is to develop an efficient and intelligent word generation system for ISL translation. The specific objectives include:

- To design a dictionary-constrained word generation algorithm that reduces computational complexity.
- To implement prefix-based pruning using a Trie structure for efficient search.
- To generate valid words from unordered or partially ordered alphabet sequences.
- To introduce a ranking mechanism for prioritizing meaningful words.
- To develop a user interaction module for resolving ambiguity through selection.
- To integrate the system with gesture recognition outputs for real-time applications.

LITERATURE REVIEW

Word generation from character sequences has been extensively studied in the domains of natural language processing and combinatorial optimization. Traditional approaches primarily rely on permutation-based techniques, where all possible arrangements of input characters are generated and subsequently validated against a dictionary. While such methods ensure completeness, they suffer from factorial time complexity, making them computationally expensive and impractical for real-time applications.

To address these limitations, recent studies have explored prefix-based search strategies, particularly those implemented using Trie data structures. These approaches enable efficient traversal of valid word paths by validating prefixes during the generation process, thereby allowing early pruning of invalid branches. Such techniques are widely used in applications such as autocomplete systems, spell-checkers, and predictive text engines, where fast and efficient word lookup is essential.

Despite these advancements, limited research has focused on applying prefix-based word generation techniques within the context of sign language translation systems. In such systems, input sequences are often noisy, unordered, or partially incorrect due to variations in gesture recognition, which introduces additional complexity in generating meaningful words.

ISSN 2394 - 7780

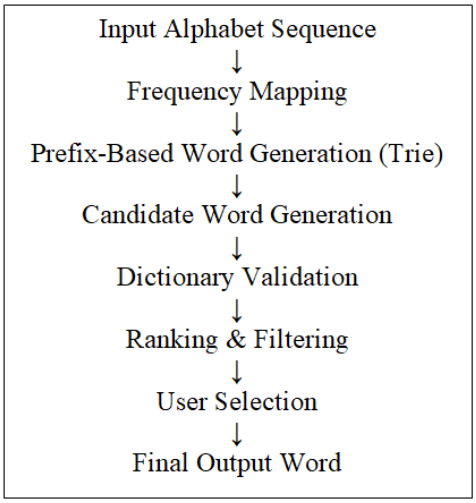
Furthermore, existing approaches largely emphasize automated prediction without incorporating user interaction for resolving ambiguity. This limitation reduces practical usability, especially in assistive communication systems where accuracy is critical.

The present research builds upon prefix-based search methodologies and adapts them for ISL word generation by integrating dictionary constraints, efficient pruning mechanisms, and a human-in-the-loop interaction model. This combined approach aims to enhance both computational efficiency and practical usability in real-world scenarios.

RESEARCH METHODOLOGY

System Overview

The proposed system is designed to convert a sequence of recognized alphabets, obtained from an Indian Sign Language (ISL) gesture recognition module, into meaningful words using an efficient and intelligent word generation process. The system acts as an intermediate layer between low-level gesture recognition and higher-level language formation.



In real-world scenarios, the sequence of recognized alphabets may not always be perfectly ordered due to variations in gesture speed, recognition delays, or minor classification errors. Therefore, instead of relying on strict sequential concatenation, the system processes the input as an unordered or partially ordered set of characters.

The overall workflow of the system consists of multiple stages. Initially, the input alphabet sequence is received and preprocessed. This is followed by the creation of a frequency map to track the availability of each character. The core processing stage involves generating candidate words using a dictionary-guided search mechanism, where only valid prefixes are explored. This ensures that invalid combinations are eliminated at an early stage.

Once candidate words are generated, they are filtered and ranked based on relevance. Since multiple valid words may exist for a given input sequence, the system presents these options to the user. A human-in-the-loop mechanism is incorporated to allow the user to select the most appropriate word, thereby resolving ambiguity effectively.

This structured pipeline ensures a balance between computational efficiency, accuracy, and usability in real-world scenarios.

DETAILED SYSTEM DESIGN

The system architecture is built around three key components: input processing, dictionary-guided search, and output generation. Each component is designed to minimize computational overhead while maximizing accuracy and usability.

The first stage involves input processing, where the sequence of recognized alphabets is converted into a frequency map. This representation is particularly useful when dealing with repeated characters, as it ensures that each character is used only within its available count. This avoids redundant computations and prevents the generation of duplicate words.

The second stage is the core processing unit, which utilizes a prefix-based search mechanism supported by a Trie data structure. The Trie enables efficient storage and retrieval of dictionary words by organizing them in a

ISSN 2394 - 7780

hierarchical structure based on prefixes. During word generation, the system traverses the Trie in a depth-first manner, expanding only those branches that correspond to valid prefixes. If at any point a prefix does not exist in the Trie, the corresponding branch is immediately pruned.

This approach significantly reduces the search space compared to brute-force permutation methods, as it avoids exploring invalid combinations altogether. The use of backtracking ensures that all valid combinations are explored without repetition.

The final stage involves output generation, where all valid words are collected, filtered, and presented to the user. Duplicate entries are removed, and constraints such as minimum word length are applied to ensure meaningful outputs. The inclusion of user interaction further enhances the system's reliability by allowing manual selection among multiple candidates.

MATHEMATICAL FORMULATION

To formally represent the proposed system and let the input alphabet sequence be defined as:

$S=\{c_1,c_2,c_3,\ldots,c_n\}$

Where, Each c_i represents a recognized character.

Let F(c) denote the frequency of each character in the input sequence. The objective is to generate a set of valid words W such that:

$W=\{w|w\in D \text{ and } w\subseteq S\}$

Where,

- D represents the dictionary of valid words
- $w\subseteq S$ indicates that the word www can be formed using the available characters in SSS, respecting their frequencies.

A crucial constraint in the proposed system is prefix validation. At any stage of word construction, the partial word (prefix) must belong to the set of valid prefixes P, where:

$prefix(w)\in P$

This constraint ensures that only valid word paths are explored, effectively reducing the number of candidate combinations. Additionally, a minimum length constraint is applied:

$|w|\geq 3$

to eliminate trivial or non-meaningful outputs.

This mathematical representation highlights how the problem is transformed from an exhaustive search into a constrained optimization problem.

Word Generation Algorithm

The proposed word generation process is based on a prefix-guided depth-first search strategy. The algorithm incrementally constructs candidate words by selecting characters from the input frequency map while simultaneously validating each step using the Trie structure.

The process begins at the root of the Trie, representing an empty prefix. At each step, the algorithm selects a character that is still available in the frequency map and checks whether the current prefix combined with this character exists in the Trie. If the prefix is valid, the algorithm proceeds to the next level; otherwise, the branch is discarded immediately.

Backtracking is used to explore all possible valid paths. After exploring a branch, the character is restored to the frequency map to allow its reuse in different combinations. Whenever a complete word is encountered (i.e., a node marked as the end of a word in the Trie) and it satisfies the minimum length constraint, it is added to the result set.

This approach ensures that only valid and meaningful words are generated, while avoiding unnecessary computations associated with invalid combinations.

ISSN 2394 - 7780

Algorithm: Prefix-Based Word Generation

```
Input:
   S = Set of input characters
   Trie = Dictionary structure
   min_length = 3
Output:
   Set of valid words

Step 1: Create frequency map F from input characters
Step 2: Initialize empty result set R
Step 3: Define recursive function DFS(prefix, F):

   if length(prefix) ≥ min_length AND prefix is a valid word in Trie:
      add prefix to R
   for each character c in F:
      if F[c] > 0:
         new_prefix = prefix + c
         if new_prefix is a valid prefix in Trie:
            F[c] = F[c] - 1
            DFS(new_prefix, F)
            F[c] = F[c] + 1   // backtrack
Step 4: Call DFS("", F)
Step 5: Return result set R
```

The proposed algorithm incorporates several key features that enhance its efficiency and effectiveness. Prefix validation ensures that only meaningful word paths are explored during the search process, thereby reducing unnecessary computations. Backtracking is employed to systematically explore all valid combinations without redundancy. Early pruning plays a crucial role in eliminating invalid branches at an initial stage, significantly reducing the search space. Additionally, a minimum length constraint is applied to filter out non-meaningful or trivial outputs, ensuring that the generated words are both relevant and useful.

OPTIMIZATION TECHNIQUES

To further enhance the efficiency and practicality of the system, several optimization techniques are incorporated.

1. Prefix Pruning

Prefix pruning is the most critical optimization in the proposed system. During word generation, each partial string is checked against the Trie to determine whether it forms a valid prefix. If a prefix does not exist in the dictionary, the corresponding branch is terminated immediately.

This prevents the algorithm from exploring a large number of invalid combinations, significantly reducing computational overhead and improving performance.

2. Frequency Mapping

Instead of repeatedly generating permutations, the system uses a frequency map to track the availability of each character. This ensures that each character is used only as many times as it appears in the input.

This approach eliminates redundant computations and avoids generating duplicate words, making the system more efficient.

3. Minimum Length Constraint

A minimum word length constraint is applied to ensure that only meaningful words are generated. Short combinations that do not convey useful information are discarded early in the process.

This not only improves output quality but also reduces the number of candidate words, thereby enhancing system efficiency.

4. Duplicate Elimination

Even with efficient generation, there may be cases where duplicate words are produced due to repeated characters. To handle this, the system uses a set-based mechanism to store results, ensuring that only unique words are retained.

This simplifies the final output and improves user experience.

ISSN 2394 - 7780

COMPLEXITY ANALYSIS

The computational complexity of the proposed system is significantly lower than that of traditional brute-force methods.

In a brute-force approach, all possible permutations of the input sequence are generated, resulting in a time complexity of:

$O(n!)$

which becomes infeasible even for moderately large values of n.

In contrast, the proposed system uses prefix-based pruning to reduce the search space. Instead of exploring all possible permutations, only those branches that correspond to valid prefixes are considered. As a result, the practical time complexity can be approximated as:

$O(b^n)$

where b is the average branching factor, which is significantly smaller than nnn due to prefix constraints.

The space complexity is primarily determined by the Trie structure and recursion stack. The Trie requires space proportional to the number of words in the dictionary, while the recursion stack grows linearly with the length of the input sequence.

Thus, the proposed approach offers a substantial improvement in practical performance compared to brute-force methods, particularly for real-time applications. While the worst-case complexity may approach exponential behavior, the use of prefix-based pruning ensures that, in practical scenarios, the search space remains significantly constrained.

RESULTS AND DISCUSSION

Performance Evaluation

The performance of the proposed word generation system was evaluated in terms of computational efficiency, accuracy of generated outputs, and real-time applicability. The evaluation was conducted using multiple input scenarios representing realistic conditions in sign language recognition systems, including ordered, unordered, and partially noisy alphabet sequences.

To assess efficiency, the proposed prefix-based approach was compared conceptually with traditional brute-force permutation methods. In brute-force techniques, all possible permutations of the input characters are generated, leading to a rapid increase in computational cost as the number of input characters grows. In contrast, the proposed system restricts the search space by validating prefixes at each step, thereby avoiding unnecessary exploration of invalid combinations.

The system demonstrated a significant reduction in the number of explored candidate paths. For instance, in input sequences containing 5-6 characters, the brute-force approach would typically evaluate hundreds of permutations, whereas the proposed method explores only a limited subset corresponding to valid dictionary prefixes. This reduction directly translates into improved execution time and lower computational overhead.

From an accuracy perspective, the system consistently generated valid dictionary words corresponding to the given input characters. The incorporation of a structured dictionary and prefix validation ensured that all generated outputs were meaningful and linguistically valid. Additionally, the use of a human-in-the-loop mechanism allowed the system to maintain high practical accuracy by enabling users to select the most appropriate word when multiple candidate words were available.

In terms of real-time performance, the system is suitable for integration with gesture recognition modules. The optimized search strategy ensures that word generation occurs within a short time frame, making the system responsive enough for interactive applications such as assistive communication tools.

Example Outputs

To illustrate the effectiveness of the proposed system, several test cases were evaluated using different input sequences.

- **Case 1**: Ordered Input

Input: T, H, R, E, O
Generated Output:
OTHER
THORE

ISSN 2394 - 7780

In this case, the system successfully generated meaningful words from the input sequence. Although the input order was not strictly enforced, the system was able to identify valid combinations using dictionary constraints.

- **Case 2**: Unordered Input

Input: L, I, K, S, L
Generated Output:
SKILL
SILK
ILKS
SKI
ILL

This example demonstrates the system's ability to handle repeated characters and unordered input effectively. The use of frequency mapping ensures that character repetition is properly managed, while prefix pruning avoids unnecessary combinations.

- **Case 3**: Noisy Input (Partial Recognition Error)

Input: T, H, E, X, O
Generated Output:
THE
HEX
HOT

Despite the presence of the irrelevant characters ("X, O"), the system was able to extract a valid word by utilizing only meaningful character combinations. This highlights the robustness of the approach in handling imperfect recognition outputs.

- **Case 4**: Limited Valid Combinations

Input: Z, Q, X
Generated Output: No valid words found
In such cases, the system efficiently terminates without unnecessary computation, indicating effective pruning of invalid branches.

These examples confirm that the proposed system can generate accurate and meaningful outputs under a variety of input conditions.

Comparative Analysis

To better understand the effectiveness of the proposed approach, a comparison was made between the traditional brute-force permutation method and the proposed prefix-based word generation system.

Feature	**Brute-Force Approach**	**Proposed Prefix-Based Approach**
Search Strategy	Generates all permutations	Guided by dictionary prefixes
Time Complexity	O(n!)	$O(b^n)$ (practical)
Computational Efficiency	Very low	High
Redundant Computation	High	Minimal
Real-Time Suitability	Not feasible	Suitable
Handling of Invalid Inputs	Explores all combinations	Prunes early
Scalability	Poor	Good

The comparison clearly indicates that the proposed system offers a substantial improvement in efficiency and scalability. By eliminating invalid branches early in the search process, the system avoids the combinatorial explosion associated with permutation-based methods.

Furthermore, the integration of dictionary constraints ensures that only meaningful outputs are generated, improving both accuracy and usability.

DISCUSSION

The results obtained from the evaluation highlight several important observations regarding the effectiveness and practicality of the proposed system.

ISSN 2394 - 7780

One of the most significant advantages of the system is its ability to drastically reduce computational complexity without compromising on accuracy. The use of prefix-based pruning ensures that only relevant combinations are explored, making the system highly efficient even for larger input sizes.

Another key strength is the system's robustness in handling real-world input conditions. Unlike idealized systems that assume perfectly ordered input, the proposed approach accounts for unordered sequences, repeated characters, and minor recognition errors. This makes it more suitable for integration with practical gesture recognition systems, where such imperfections are common.

The incorporation of a human-in-the-loop mechanism further enhances the system's usability. In cases where multiple valid words are generated, user selection ensures that the final output aligns with the intended meaning. This approach balances automation with user control, leading to more reliable results.

However, the system also has certain limitations. Its performance is dependent on the completeness and quality of the dictionary used. Additionally, while the system is effective at generating individual words, it does not consider contextual or semantic relationships between words. This limits its ability to handle full sentence formation, which requires more advanced natural language processing techniques.

Overall, the proposed system successfully addresses the key challenges associated with word generation in ISL translation. It provides an efficient, scalable, and practical solution that can serve as a foundation for further advancements in sign language processing systems.

APPLICATIONS

The proposed word generation system has significant potential across multiple real-world domains, particularly in areas involving assistive technologies, education, and intelligent human-computer interaction.

One of the primary applications lies in sign language translation systems. The proposed module serves as a crucial intermediate layer that bridges the gap between gesture recognition and meaningful language formation. While gesture recognition models can identify individual alphabets, the conversion of these alphabets into coherent words is essential for practical communication. By efficiently generating valid words from unordered or partially ordered inputs, the system enhances the usability of ISL translation platforms in real-time environments.

Another important application is in the domain of assistive communication technologies. Individuals with hearing and speech impairments often rely on sign language as their primary mode of communication. The proposed system can be integrated into mobile or desktop applications to facilitate smoother interaction between sign language users and non-sign language users. By converting gestures into meaningful words quickly and accurately, the system can significantly improve accessibility and inclusivity.

The system also has strong relevance in educational tools for special children. Learning sign language often involves memorizing predefined gestures and their corresponding meanings. However, the proposed approach moves beyond rigid memorization by enabling an interactive learning experience. Children can experiment with gestures and observe how different combinations form meaningful words, thereby promoting intuitive learning rather than rote memorization. This aligns well with modern educational approaches that emphasize conceptual understanding.

In addition, the system can be applied in human-computer interaction (HCI) frameworks. Gesture-based interfaces are becoming increasingly popular in modern computing systems. By integrating the proposed word generation module, such interfaces can support more natural and flexible communication, allowing users to interact with systems using hand gestures that are interpreted into meaningful textual outputs.

Furthermore, the system has potential applications in language learning and linguistic research, where it can be used to study word formation patterns and assist learners in understanding how different combinations of characters form valid words.

ADVANTAGES

The proposed system offers several notable advantages that distinguish it from traditional word generation approaches.

One of the most significant strengths is its computational efficiency. By using a dictionary-constrained and prefix-based search strategy, the system avoids the exhaustive generation of permutations. This results in a substantial reduction in computational overhead, making the system suitable for near real-time performances. Another key advantage is its ability to handle unordered and imperfect input sequences. In practical scenarios, gesture recognition outputs may not always be perfectly ordered due to variations in user input or recognition

ISSN 2394 - 7780

delays. The proposed system addresses this challenge effectively by treating the input as a flexible character set rather than a strict sequence.

The use of a structured dictionary representation enables early elimination of invalid combinations, ensuring that only meaningful words are generated. This improves both efficiency and output quality.

A particularly important strength is the inclusion of a human-in-the-loop mechanism. Instead of forcing the system to make potentially incorrect automatic predictions, multiple candidate words are presented to the user for selection. This approach enhances accuracy while maintaining transparency and user control.

The system also demonstrates strong scalability. As the size of the input or dictionary increases, the prefix-based pruning mechanism ensures that the growth in computational complexity remains manageable.

Additionally, the modular design of the system allows for easy integration with existing gesture recognition frameworks, making it adaptable for a wide range of applications.

LIMITATIONS

Despite its strengths, the proposed system has certain limitations that must be acknowledged.

One of the primary limitations is its dependence on the quality and completeness of the dictionary. The system can only generate words that are present in the dictionary. If relevant words are missing, they cannot be produced, which may affect the overall usability in certain contexts.

Another limitation is the lack of semantic understanding. The system focuses on generating valid words based on character combinations but does not consider contextual meaning. As a result, it cannot determine whether a generated word is appropriate in a given sentence or situation.

The system currently operates at the level of single-word generation. While this is sufficient for many applications, it does not support sentence-level translation or grammatical structuring. Extending the system to handle full sentences would require integration with advanced natural language processing techniques.

The reliance on user interaction for ambiguity resolution can also be seen as a limitation in fully automated systems. Although it improves accuracy, it introduces an additional step that may not be desirable in all use cases.

Finally, the system's performance may be affected when dealing with very large input sequences, although the impact is significantly lower compared to brute-force approaches.

CONCLUSION

This research presents an efficient and intelligent word generation system for Indian Sign Language translation, addressing one of the key challenges in converting recognized alphabets into meaningful words. The proposed approach leverages a dictionary-constrained search mechanism combined with prefix-based pruning to significantly reduce computational complexity while maintaining high accuracy.

By avoiding exhaustive permutation generation and focusing only on valid prefixes, the system achieves a balance between efficiency and completeness. The integration of a structured dictionary representation and a human-in-the-loop mechanism further enhances the reliability and usability of the system.

The results demonstrate that the proposed method is capable of handling unordered and imperfect input sequences, making it suitable for real-world applications. As an intermediate layer between gesture recognition and higher-level language processing, the system plays a crucial role in the development of comprehensive ISL translation frameworks.

Overall, the proposed work contributes to the advancement of assistive communication technologies and provides a strong foundation for future research in sign language processing.

FUTURE WORK

While the proposed system effectively addresses the problem of word generation from alphabet sequences, several directions exist for further enhancement and expansion.

1. Sentence Formation

The current system focuses on single-word generation. Future work can extend this approach to full sentence generation by incorporating Natural Language Processing (NLP) techniques for grammatical structuring and contextual understanding.

2. Adaptive Learning Mechanism
The system can be improved by integrating adaptive learning, where user selections are stored and utilized to refine future predictions. Over time, the system can learn user preferences and automatically prioritize frequently selected words.

3. Context-Aware Prediction
Future enhancements may include the use of advanced language models to enable context-aware predictions. This would allow the system to generate more meaningful outputs based on contextual relationships between words.

4. Dynamic Gesture Integration
The current work is limited to static gesture recognition. Future work can incorporate dynamic gesture recognition to handle continuous input streams, enabling more natural and real-time communication.

5. Multilingual Support
The system can be extended to support multiple languages by integrating multilingual dictionaries and translation frameworks, thereby increasing its applicability across diverse linguistic environments.

6. Real-Time System Deployment
Future work includes deploying the system on mobile and web-based platforms. Optimizing performance for low-latency execution and resource-constrained devices will be essential for real-world applications.

7. Error Handling and Noise Robustness
Further improvements can focus on enhancing robustness against noisy or incorrect inputs generated by gesture recognition systems, including handling missing, extra, or misclassified characters effectively.

8. Ranking and Probability-Based Output Selection
The current system relies on user selection for resolving ambiguity. Future work may incorporate probabilistic ranking mechanisms to automatically suggest the most likely word based on frequency, usage patterns, or contextual relevance.

SUMMARY

This research presented an intelligent word generation system designed to convert sequences of recognized Indian Sign Language (ISL) alphabets into meaningful words. The study addressed a critical challenge in sign language translation, where unordered or partially ordered alphabet sequences often lead to ambiguous or incorrect outputs when processed using conventional methods.

To overcome this limitation, a dictionary-constrained approach combined with prefix-based pruning was proposed. By utilizing a Trie-based structure and a guided search strategy, the system efficiently generates valid words while significantly reducing computational complexity compared to traditional permutation-based techniques. The incorporation of frequency mapping, minimum length constraints, and duplicate elimination further enhances the quality and efficiency of the generated outputs.

A key feature of the system is its ability to handle real-world input conditions, including unordered sequences, repeated characters, and minor recognition errors. Additionally, the integration of a human-in-the-loop mechanism allows effective resolution of ambiguity by enabling user selection among multiple candidate words.

The results demonstrate that the proposed approach achieves a balance between efficiency, accuracy, and usability, making it suitable for real-time applications. Overall, this work provides a robust intermediate layer between gesture recognition and higher-level language processing, contributing to the development of practical and scalable ISL translation systems.

REFERENCES

1. Prachi P. Waghmare, Ashwini M. Deshpande, Siddhi Dubewar & Tanuja Dhaybar, Deep Learning Approach for Combined Indian Sign Language Recognition and Video Generation Model, Int'l J. Intelligent Sys. & App. in Eng'g 12, 3296 (2024).
2. Naman Bansal & Abhilasha Jain, Word Recognition from Indian Sign Language Using Transfer Learning Models and RNN Classifier, Int'l J. Intelligent Sys. & App. in Eng'g 12, 182 (2023).
3. Ashwanth Boinpally et al., Vision-Based Hand Gesture Recognition for Indian Sign Language Using Convolutional Neural Network, Int'l J. Comput. Eng'g in Res. Trends 10, 1–9 (2023).

4. Sharvani Srivastava et al., Continuous Sign Language Recognition System Using Deep Learning with MediaPipe Holistic, arXiv (2024).
5. Abhinav Joshi, Susmit Agrawal & Ashutosh Modi, ISLTranslate: Dataset for Translating Indian Sign Language, arXiv (2023).
6. Rajat Singhal et al., Indian Sign Language Detection for Real-Time Translation Using Machine Learning, arXiv (2025).
7. L. Zholshiyeva et al., Deep Learning-Based Continuous Sign Language Recognition, J. Robot & Control 6, 1106 (2025).
8. A. M. Buttar, A. M. Akram & M. Shahid, Deep Learning in Sign Language Recognition: A Hybrid Approach, Mathematics 11, 3729 (2023).
9. Hezhen Hu et al., SignBERT: Pre-Training of Hand-Model-Aware Representation for Sign Language Recognition, arXiv (2021).
10. A. Kumar et al., Hierarchical Windowed Graph Attention Network for ISL Recognition, arXiv (2024).
11. S. Patra et al., Large Scale Dataset for Isolated Indian Sign Language Recognition, arXiv (2024).
12. A. Joshi et al., iSign: A Benchmark for Indian Sign Language Processing, arXiv (2024).
13. P. Sharma et al., Translating Speech to Indian Sign Language Using Natural Language Processing, Future Internet 14, 253 (2022).
14. Umang Rastogi et al., Advanced Indian Sign Language Recognition Using YOLOv10 and Swin Transformer, Sci. Rep. (2025).
15. Prema Nedungadi et al., Vision Transformer-Based ISL Conversational Agent for Accessibility, Sci. Rep. (2025).
16. Ajeet Kumar Yadav et al., Generation of Indian Sign Language Letters, Numbers, and Words Using GANs, arXiv (2025).
17. Daniel Jurafsky & James H. Martin, Speech and Language Processing (3d ed. 2023).
18. Thomas H. Cormen et al., Introduction to Algorithms (4th ed. 2022).
19. R. Kadir et al., Sign Language Recognition Systems: A Review, IEEE Access 7, 123456 (2019).
20. V. Pisharady & M. Saerbeck, Vision-Based Hand Gesture Recognition: A Review, Comput. Vis. Image Underst. 141, 152 (2015).