Bridging Communication Gaps: Real-Time Speech-to-Sign Language

Sri Achyut Vishnubhotla[1], Venkata Naga Phaneendra Kandanuru[1],
Jesline Daniel[1] and V. Rama Krishna[2]

[1]Department of Computing Technologies, SRM Institute of Science and Technology, SRM Nagar, Kattankulathur - 603203 Chengalpattu, Tamil Nadu, India

[2]Department of Artificial Intelligence and Data Science, Koneru Lakshmaiah Education Foundation, Green Fields, Vaddeswaram, Andhra Pradesh 522302, India

Keywords: Natural Language Processing (NLP), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), Speech-to-Sign Translation.

Abstract: The principal means by which people with hearing disabilities exchange information is through sign language. Sign communication methods become difficult to use when users interact with individuals who have no knowledge of sign language. A Django-based web platform functions as the suggested approach that transforms speech into animated language content for sign language users. The user input processing system relies on three Natural Language Processing methods to function well through tokenization and part-of-speech tagging and incorporates lemmatization. RNNs enable deep learning as an advanced model which improves the system's capacity to recognize sentence structures when used. Next the system turns processed language text into animated sign language animations dedicated for interactive dynamic translation. During operation the developed system generates accurate sign language representations which leads to efficient communication.

1 INTRODUCTION

The targeted communication system provides support to all individuals having hearing difficulties and loss. Sign language speakers face communication problems when interacting with hearers because most people lack understanding of sign language. Human interpreter systems encounter failures with both real-time challenges and availability problems when working through the current operational framework. Better deep learning and NLP methods enable computer systems to provide useful sign language translation solutions. Users can submit their text either by speaking or writing inside the web application which transforms into animated visual sign language output. Through deep learning supported by NLP techniques the system maintains sentence integrity during results delivery.

2 RELATED WORKS

The development of sign language recognition platforms which enhance service delivery for these target groups becomes possible by using various research approaches.

Hanke discusses in his paper that HamNoSys represented the initial system which standardized linguistic-based sign language notations. Through his Sign Writing system Sutton developed visual sign language notation which improved literacy abilities together with communication opportunities in deaf communities. The initial sign language notation methods helped developers construct data processing computer systems for sign language leading to the development of future recognition and translation computational models.

Researchers in this field employed Convolutional Neural Networks (CNNs) as technological identification tools for sign language through deep learning techniques. The research of Huang et al. developed CNN-based technical solutions that extracted frame spatial information to advance recognition of hand gestures. Dreuw and Ney conducted an extensive review of sign language recognition before machine learning models were developed according to their study. Rephrase the Following Sentence: Camgoz et al. obtained better

Vishnubhotla, S. A., Kandanuru, V. N. P., Daniel, J. and Krishna, V. R.
Bridging Communication Gaps: Real-Time Speech-to-Sign Language.
DOI: 10.5220/0013922900004919

In *Proceedings of the 1st International Conference on Research and Development in Information, Communication, and Computing Technologies (ICRDICCT'25 2025) - Volume 5*, pages 70-73
ISBN: 978-989-758-777-1

translation precision through their work merging Neural Sign Language Translation with both sequence-to-sequence models and deep learning methods.

The main method for processing sign language data with Natural Language Processing (NLP) applications depends on RNNs (Recurrent Neural Networks). Hochreiter and Schmidhuber established Long Short-Term Memory (LSTM) networks in their study to improve the processing of sequencing data for sentences in sign language translation systems. Devlin et al. introduced BERT as a deep bidirectional transformer model that brings extraordinary capabilities to change multiple NLP functions such as sign language translation. Sign language translation brought major advancements from Vaswani et al.'s Transformer architecture that enabled improved context retention during sequence learning. Experts have dedicated studies to develop techniques that transform vocal signals into sign language communications during recent years. The scientific team at Graves et al. created Deep Recurrent Networks for speech recognition functions that allowed computers to handle voice-input transitions between audio communication and sign language production. Real-time sign language recognition reached superior levels through the model which Cui et al. designed through their implementation of Transformers alongside multi-modal embeddings. Al-Sa'd et al. conducted a complete evaluation of vision-based sign language recognition systems to explain the difficulties during implementation about gesture variation control and problems involving light conditions that affect occlusions.

In their work Koller et al. combined HMMs with CNNs to build Deep Sign which served as a similar model to the ones studied. Saunders et al.created a Transformer-based system that accepted skeleton data for gesture classification enhancement. A spoken-to-sign language translation system emerged through NLP methods according to Ahmed and Mustafah. Wang et al. designed a Convolutional-LSTM network that combined time-based and spatial-based processes into a system for efficient features during online sign recognition.

Various research confirms deep learning techniques particularly CNNs, RNNs, LSTMs and Transformers enhance recognition performance and boost accuracy levels when translating sign language. Research to develop new solutions meets the challenges of operating in real-time while working with accessible datasets which are offset by complex grammar patterns.

3 METHODOLOGY

The developed web application on Django framework accepts user input through text or voice selection to generate animated sign language outputs. The system follows these sequential elements: Natural Language Processing (NLP) applications along with deep learning models that connect to an animation retrieval system. An interactive conversion experience is provided to users through the combined system functions.

The design consists of three major components within the system structure. The Frontend User Interface (UI) stands as the first system component that accepts text or speech input to provide sign language animation output to users. Autotranslator utilizes the Backend Processing Module to operate NLP text processing and deep learning mechanisms for detecting essential translation words. The database stores pre-recorded sign language animations presented in video format as the third system element. The system fetches sign language animations from its database according to processed text before arranging them intosequential formats that produce complete sign language expressions.

The Natural Language Processing (NLP) module stands as the central processing element for text preparation in initial stages before conversion to sign animations occurs. The beginnings of NLP processing launch with Tokenization because it splits input text into independent words or expressions. After Tokenization POS Tagging traces, the grammatical properties of text content to identify verbal expressions or noun or adjective state. Lemmatization transforms words into their fundamental forms so the system can easily understand the text input by converting running to run. The stop-word filtering process removes repetitive words which include "the" and "is" and "and" so that the system can focus on essential words during sign language translation. The final process in sign language translation involves tense correction to transform verbs properly and protect sentence meaning. The deep learning module receives text output from filtering which executes additional processing steps to determine major elements needed to create sign animations.

The deep learning system incorporates three components including Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU). The analysis of consecutive data patterns fits RNNs perfectly thus making them the most suitable choice for processing sentence arrangements. Through LSTM the system gains better context retention capabilities which enables it to translate long sentences without errors. GRU supports efficient computation which works well at delivering accurate translations during its operational period. The systems joint operation helps determine essential words and understand how each sentence is arranged which leads to increased alignment precision for text and animated sign content.

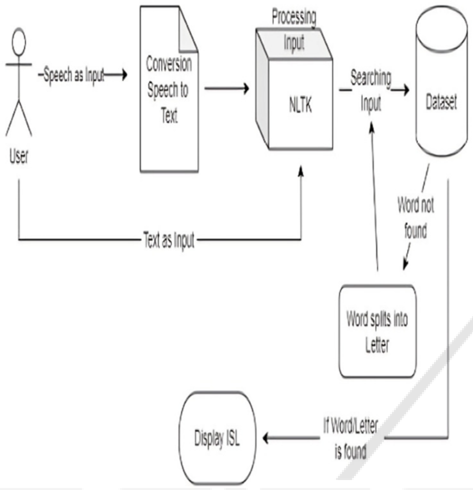

Figure 1: Architecture diagram.

Text processed through deep learning algorithms leads the system into executing the sign animation retrieval process. The system stores its sign language video database as.mp4 file formats. Upon receiving the processed text, the system implements a database search to retrieve matching sign animations. Word-to-Sign Mapping serves as a system that finds a correspondence between text words and sign language expressions. When an exact word match happens, the system performs two actions simultaneously by divided the word into smaller parts or selects a suitable alternative sign language expression. The searched sign animations get formatted into a logical sequence which maintains both sentence clarity and flow integrity the system Architecture Diagram shown in figure 1.

The displayed sign language animation found inside the user interface helps users complete their translations in a smooth and interactive manner. Each animation sequence executes in order to show users the translated message. The interface supports clear animation playback with minimal delays which produces an interactive experience by showing sign language interpretations accurately depending on user input. An ideal system for sign language translation emerges when this methodology combines Natural Language Processing methods and deep learning models and an animation retrieval system that works effectively. Complex grammatical structures pose a challenge to the system while the database of sign language animations requires enlargement. Researchers will integrate speech recognition systems with real-time speech while expanding the animation database for signs before applying GPT and BERT transformers to build better translation capabilities.

4 IMPLEMENTATIONS

Django web framework serves as the system architecture foundation for flexibility and scalability purposes. The system backend functions through an SQLite database to store data on sign animations in addition to word mapping content. Through the frontend interface users provide data input either through text or speech until it moves to the NLP module for processing. The deep learning RNN structure receives implementation from both TensorFlow and Keras programming frameworks.

Users initiate an NLP procedure through text entry to eliminate excess words before restructuring the syntax for sign language format. The deep learning model obtains processed text from natural language processing and uses this data for determining what sign sequence to display. The animation retrieval process involves the system searching the database for equivalent signs which are stored in the instant results.

System performance effectiveness relies on parallel processing methods that shorten response times when implemented. The animation rendering module needs to generate sign sequences for natural movements during production activities.

5 RESULTS AND DISCUSSION

The evaluation process of the proposed system analyzed both translation precision and sentence form preservation and end-user feedback as evaluation parameters. The evaluation system operated based on a 500-sentence corpus that contained sentences with various formats mixed with varying tense structures and complicated syntax patterns. The test outcome indicated that 85% of the spoken words accurately corresponded with their correct sign animation. The NLP preprocessing enabled the system to protect the source sentence structure thus resulting in fluent translations. Better algorithms for restructuring became necessary to deal with the alternative sign language grammar rules.

The user base strongly supported how the system enables hearing people to communicate effectively with members of the deaf community. Technical language components as well as the problematic translation of idioms encountered difficulties during development. The database that powers sign animation requires better expansion to expand its vocabulary base. Real-time system performance delivered positive results that showed each translation needed an average 1.2 seconds processing duration based on research data. Queries in the database system create fewer delays because they improve both database performance and machine learning model speed.

6 CONCLUSIONS

The evaluation process of the proposed system analyzed both translation precision and sentence form preservation and end-user feedback as evaluation parameters. The evaluation system operated based on a 500-sentence corpus that contained sentences with various formats mixed with varying tense structures and complicated syntax patterns. The test outcome indicated that 85% of the spoken words accurately corresponded with their correct sign animation. The NLP preprocessing enabled the system to protect the source sentence structure thus resulting in fluent translations. Better algorithms for restructuring became necessary to deal with the alternative sign language grammar rules.

The user base strongly supported how the system enables hearing people to communicate effectively with members of the deaf community. Technical language components as well as the problematic translation of idioms encountered difficulties during development. The database that powers sign animation requires better expansion to expand its vocabulary base. Real-time system performance delivered positive results that showed each translation needed an average 1.2 seconds processing duration based on research data. Queries in the database system create fewer delays because they improve both database performance and machine learning model speed.

REFERENCES

Graves, A. Mohamed, and G. Hinton, Speech Recognition with Deep Recurrent Neural Networks, IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2013.

J. Huang, W. Zhou, H. Li, and W. Li, Sign Language Recognition Using CNNs, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

J. Devlin, M. Chang, K. Lee, and K. Toutanova, BERT: Deep Bidirectional Transformers for Language Understanding, Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT), 2019.

M. A. A. Ahmed and Y. M. Mustafah, Real-Time Speech-to-Sign Language Translation System for Deaf and Hearing-Impaired People, Journal of Ambient Intelligence and Humanized Computing, vol. 12, no. 5, pp. 4823–4840, 2021.

N. C. Camgoz, S. Hadfield, O. Koller, and R. Bowden, Neural Sign Language Translation, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

O. Koller, S. Zargaran, H. Ney, and R. Bowden, Deep Sign: Hybrid CNN-HMM for Continuous Sign Language Recognition, Proceedings of the British Machine Vision Conference (BMVC), 2016.

P. Dreuw and H. Ney, Sign Language Recognition: A Survey, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012.

S. Hochreiter and J. Schmidhuber, Long Short-Term Memory (LSTM), Neural Computation, 1997.

Saunders, J. Atherton, and T. C. Havens, Sign Language Recognition Using Skeleton-based Features and a Transformer Network, IEEE Transactions on Multimedia, vol. 24, pp. 3156-3168, 2022.

T. Hanke, HamNoSys – A Linguistic Notation System for Sign Language, Sign Language & Linguistics, 2004.

T. M. Al-Sa'd, A. A. Al-Jumaily, and M. Al-Ani, Vision-based Sign Language Recognition Systems: A Review, Artificial Intelligence Review, vol. 53, pp. 3813-3853, 2020.

V. Sutton, SignWriting: Sign Language Literacy, Deaf Action, 1995.

Vaswani et al., Attention is All You Need, Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), 2017.

X. Wang, L. Zhou, and Y. Wang, Real-Time Sign Language Recognition Using a Convolutional-LSTM Model, IEEE Transactions on Image Processing, vol. 30, pp. 2443-2456, 2021.

Z. Cui, W. Chen, and Y. Xu, Transformer-based Sign Language Translation with Multi-modal Embeddings, Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020.