Int. J. Advance Soft Compu. Appl, Vol. 18, No. 1, March 2026
Print ISSN: 2710-1274, Online ISSN: 2074-8523

Generating Natural Language from AAC Cards for Children with Autism using a Sequence-to-Sequence System

Lely Hiryanto[1], Marchella Angelina[2], and Tony Tony[3]

[1] Faculty of Information Technology, Tarumanagara University
e-mail: lelyh@fti.untar.ac.id
[2] Faculty of Information Technology, Tarumanagara University
e-mail: marchella.535220001@stu.untar.ac.id
[3] Faculty of Information Technology, Tarumanagara University
e-mail: tony@fti.untar.ac.id

Abstract

Communication deficits are a common challenge experienced by children with Autism Spectrum Disorder (ASD). Existing Augmentative and Alternative Communication (AAC) applications, such as VICARA, are limited to unidirectional communication—only transmitting raw card sequences to another application (e.g., "I, Eat, Noodle"). This limitation renders the conversation flow non-interactive and prone to miscommunication. This study aims to develop a new version of VICARA, named Talk of the Heart AAC, a bidirectional Android-based application capable of bridging the communication gap between autistic children and their caregivers. The application is designed to translate the constructed card sequences into complete, natural, and semantically logical sentences in Bahasa Indonesia through the implementation of a Sequence-to-Sequence model with a Gated Recurrent Unit architecture augmented by an attention mechanism. To train the model, a synthetic dataset comprising 7,000 data pairs was generated using the Google Gemini API and validated by a speech-language pathology expert. Model evaluation results demonstrated a good performance in sentence translation, yielding a BLEU score of 42.95%, ROUGE-1 94.94%, ROUGE-2 87.12%, and ROUGE-L 93.70%. From the application perspective, non-functional testing produced an average System Usability Scale score of 91.67, categorized as "excellent."

Keywords: *Augmentative and Alternative Communication (AAC), Autistic Children, Gated Recurrent Unit (GRU), Sentence Translator, Sequence-to-Sequence*

1 Introduction

Deficits in both verbal and nonverbal communication are core challenges for individuals with Autism Spectrum Disorder (ASD), significantly affecting social interaction, academic performance, and child's independence [1]. These communication challenges often lead to parental stress and the emergence of challenging behaviors [2]. To bridge this critical gap, Assistive Communication Technology (ACT), particularly in the form of Augmentative and Alternative Communication (AAC) systems, has become a crucial intervention tool.

Received 21 November 2025; Accepted 11 March 2026

AAC is a therapeutic approach designed to augment or, in some cases, replace natural verbal communication. This is achieved using sign language, body language, the Picture Exchange Communication System (PECS), and speech-generating devices (SGDs) [3]. Previous studies have consistently demonstrated the effectiveness of AAC-based interventions in enhancing the communication skills, social interaction, and reducing challenging behaviors among children with autism [4].

To address the specific need for localized communication support, the Visually Interactive Communication and Reading Aid (VICARA) was developed as a Bahasa Indonesia-based AAC application. VICARA is designed to assist children with autism who have a speech disability to express their feelings and needs by simply tapping one or more picture cards that generate corresponding spoken words (speech) [1].

While the previous version of VICARA has been proved beneficial, a User Acceptance Testing (UAT) was conducted with teachers teaching special need children revealed a critical limitation in its communication flow. The prior version supports only unidirectional communication, transmitting the child's message (a sequence of raw, non-grammatical keywords generated from picture cards) externally via WhatsApp, rather than facilitating in-app interactive dialogue.

Another AAC application is TouchChat [6], [7], another example of a popular AAC application. This app offers 40,000 selectable picture cards and allows the use of personal photos to create custom cards. Similar to VICARA, TouchChat sends the results of the card arrangement, which is in raw text format, via external communication media, i.e., email dan iMessage.

This design limitation severely restricts the ability to maintain sustained, two-way communication. Since the system relies only on sending raw keywords, the full burden of semantic interpretation falls entirely on the recipient. This reliance on manual decoding not only increases the risk of misunderstandings but also creates a significant barrier to developing mutual understanding between the child and their communication partner, potentially leading to a continuous cycle of miscommunication within the family [2].

In this paper, we proposed a new version of VICARA, called "Talk of the Heart AAC" that integrates a bidirectional messaging feature built directly within the application. Our work employs sequence-to-sequence (seq2seq) model with Gated Recurrent Units (GRU) architecture to generate a complete sentence in Bahasa Indonesia, i.e., the sentence consists of subject, verb, object and adverb, by using a set of words that may missing one or more of verb, subject, object and adverb. When a child with autism arranges picture cards, a Seq2Seq model translates the raw, non-grammatical keyword sequence (e.g., "Saya (I)", "Sedih (Sad)", "Mangkuk (Bowl)", "Kotor (Dirty)") into a fluent and natural sentence for their parents (e.g., "Saya merasa sedih karena mangkuk terlihat kotor (I feel sad because the bowl looks dirty)."). Furthermore, the system supports reciprocal communication by enabling parents to reply using picture cards, which are visually understood by the child.

2 Related Work

The application of sequence-to-sequence (Seq2Seq) models using deep learning methods like Long Short Term Memory (LSTM), Recurrent Neural Networks (RNN), and Gated Recurrent Units (GRU) has become widely popular for generating text [2]. Henry *et.al.* [3] proposed Seq2Seq model based on CopyNet to automatically generate factoid questions in Bahasa Indonesia. CopyNet uses GRU and Bi-GRU as the encoder and LSTM the decoder.

Another work by Damayanti *et.al.* [4] created a set of sentences that make up a story based on a set of words as input using RNN with GRU.

This study extends two preliminary works in [1], [5] by making the following contributions:

1 Designing and implementing a novel Seq2Seq model with a Gated Recurrent Unit (GRU) encoder-decoder and attention mechanism, specifically tailored for translating non-grammatical picture cards into natural sentences in Bahasa Indonesia.
2 Developing a robust bidirectional communication system architecture within a mobile application to directly address the unidirectional limitation of previous VICARA versions.
3 Evaluating the model's performance quantitatively using standard metrics (ROUGE and BLEU) and validating the application's user experience to confirm its effectiveness in assistive technology contexts.

3 Methodology

This study employed an end-to-end model development pipeline [6], [7] to design, train, and evaluate the Seq2Seq translation system. The primary stages of the pipeline, depicted in Fig. 1, include dataset generation, data preprocessing, model architecture design, training and implementation, and quantitative evaluation.

Fig. 1: The End-to-End Model Development Pipeline

3.1 Dataset Generation

Since a large-scale public corpus for Indonesian AAC-to-text translation was unavailable, we developed a new dataset specifically for this research. Regarding the dataset type and size, we generated a synthetic dataset comprising 7,000 unique data pairs. Furthermore, the tool used for generating this dataset was the Google Gemini Large Language Model (LLM) API. The generation process was constrained by a predefined vocabulary of 155 keywords from the VICARA 3 application's picture cards, which is the latest version and is available on the Google Play Store [8]. The dataset was constructed through a two-stage generation process:

1 Raw Keyword Sequence Generation: The LLM was prompted to create logical and contextually relevant combinations of two to six keywords from the predefined vocabulary. This stage simulated a user's message composition process, with prompts designed to enforce keyword constraints, prevent internal word repetition, and mitigate semantic contradictions.
2 Complete Sentence Generation: Each valid keyword sequence from the first stage served as a new prompt, instructing the LLM to generate a complete, natural, and grammatically correct Indonesian sentence that accurately reflected the underlying semantic message, ensuring all source keywords were integrated into the final sentence.

To ensure each data pair was logical, contextually natural, and aligned with the communication needs of children on the autism spectrum, the synthetic dataset underwent expert validation. The dataset was reviewed by Siti Asma, S.Pd., founder of House of Faith

learning center and a Speech-Language Pathologist expert. The final, expert-validated dataset consists of parallel data pairs, where each raw keyword sequence (input) is directly mapped to its corresponding full sentence (target/output). Several examples from this dataset are depicted in Fig. 2.

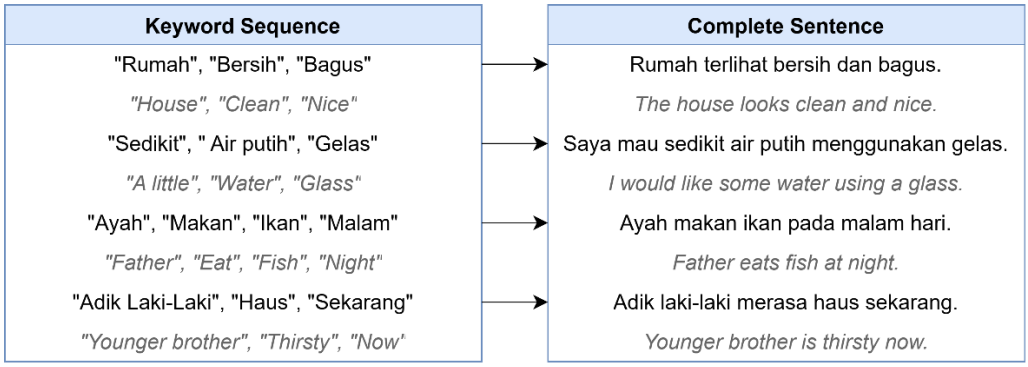

Fig. 2: Sample of Keyword-to-Sentence Translations from the Dataset

3.2 Dataset Preprocessing

The generated keyword-to-sentence dataset was then preprocessed to convert raw text into machine-readable numerical tensors suitable for the Seq2Seq model. This pipeline ensured consistent and machine-readable data:

1 Tokenization and Normalization: Both the input keyword sequences and target sentences were segmented into tokens (words/phrases) and normalized to lowercase (case folding).
2 Vocabulary Construction: Two distinct vocabularies were then constructed for the source (encoder) and target (decoder), mapping each unique token to an integer index. Out-of-vocabulary (OOV) words were mapped to the special <unk> token.
3 Special Token Integration: To guide the decoding autoregressive process, the target sentences were augmented by preprending a <start> token and appending an <end> token.
4 Padding and Target Shifting: Post-padding <pad> was applied to all sequences to ensure a uniform length within batches. Crucially, the final target sequence was shifted one timestep left to implement the teacher forcing mechanism during training.

3.3 Model Architecture

The Seq2Seq [15] that we developed uses an encoder-decoder structure enhanced with a dot-product attention mechanism, as depicted in Fig. 3. We implemented the entire architecture using the TensorFlow [16], [17] and Keras libraries [18].

1 Encoder: The encoder processes the non-grammatical input keyword sequence to generate a rich sequence of contextual hidden states (annotations). First, the input tensor passes through an Embedding layer, which transforms each token index into a 256-dimensional dense vector. The embedded sequence is then processed by a stack of two Bidirectional a Gated Recurrent Unit (GRU) layers, each uses a latent dimension of 384 units. This generates a 768-dimensional sequence of rich hidden states (a concatenation of the forward and backward states) that successfully captures the context of the entire input.

2 Attention Mechanism: The dot-product attention mechanism [9] acts as the bridge between the encoder and decoder. At every decoding timestep, this mechanism calculates alignment scores by simply comparing the decoder's current hidden state with all encoder hidden states. These scores are then normalized using a softmax function to produce attention weights, indicating input relevance for the current prediction. Finally, a context vector is computed as the weighted sum of the encoder's hidden states, providing the focused information required by the decoder.

3 Decoder: The decoder is an autoregressive component that generates the target sentence one token at a time. Its input (the previously generated token, starting with the <start> token) is first fed through its own Embedding layer (256 dimensions) and then processed by a stack of two Unidirectional GRU layers, each containing 768 units to match the dimensional output of the encoder. At each timestep, the context vector from the attention mechanism is concatenated with the decoder's final GRU output, resulting in a combined vector of 1536 dimensions. This vector is then passed into a final Dense output layer with 526 units, corresponding to the target vocabulary size and a softmax activation function, which yields the probability distribution for the next token prediction.

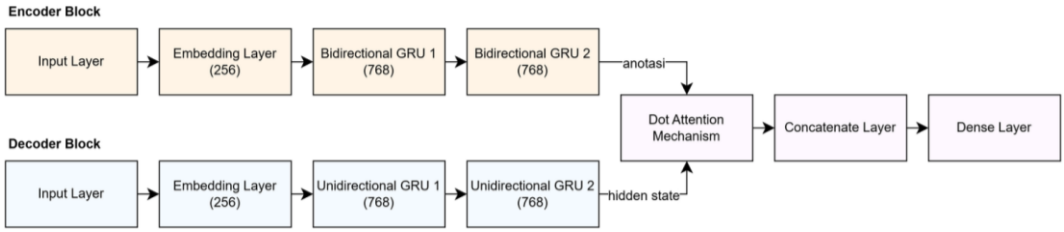

Fig. 3: Seq2Seq Model Architecture

3.4 Implementation

The model was trained on the partitioned dataset, training (80%), validation (10%), and test (10%) sets. All model layers were initialized using the default Keras Glorot Uniform scheme. The training was conducted for a maximum of 75 epochs with a batch size of 32. An Adam optimizer was utilized with an initial learning rate of 5×10^{-4}. To ensure stability during training, gradient clipping (maximum norm of 1.0) was applied to prevent the exploding gradients problem. The model was optimized using a categorical cross-entropy loss function enhanced with label smoothing (set $\alpha = 0.1$). This technique helps regularize the model and improves its generalization capabilities. To further prevent overfitting and ensure robust training, a set of callbacks was utilized:

1 EarlyStopping monitored validation loss and restored best weights if no improvement was seen for 8 consecutive epochs.
2 ReduceLROnPlateau adaptively reduced the learning rate by a factor of 0.5 if the validation loss plateaued for 4 epochs.
3 ModelCheckpoint saved the model achieving the lowest validation loss.

3.5 Quantitative Evaluation

To evaluate the generalization capability of the final trained model, translation quality was quantitatively measured on the test set using two complementary metrics:

1 Recall-Oriented Understudy for Gisting Evaluation (ROUGE) [10], [11]: We prioritized this metric to ensure full keyword coverage and to maintain semantic integrity (the meaning) in the final translation. The individual components of this metric quantify content fidelity through recall-based overlap, quantify this requirement:
   a ROUGE-1 (Unigrams) measure recall based on the overlap of unigrams (single words) between the predicted and reference sentences.
   b ROUGE-2 (Bigrams) measure recall based on the overlap of bigrams (sequenced word pairs) between the predicted and reference sentences.
   c ROUGE-L (LCS-based) uses the concept of LCS to measure the longest common sequence of words between the predicted and reference sentences.
2 Bilingual Evaluation Understudy (BLEU) [12]: BLEU serves as a complement to ROUGE by quantifying fluency and adequacy. It measures modified n-gram precision (typically up to 4-grams) and incorporates a brevity penalty to discourage overly short outputs. The resulting high BLEU score confirms that the model's output sentences are grammatically coherent and resemble human-level fluency.

4 Seq2Seq System

To ensure the seq2seq model works effectively and efficiently on resource-constrained mobile devices, a server-side deployment strategy was adopted, as depicted in Fig. 4. Since the Seq2Seq architecture is highly computationally intensive, we decoupled the entire inference process from the Android application and moved it to an external web service. The model was deployed using the FastAPI framework and served via Uvicorn [13], [14], creating a dedicated prediction endpoint. This endpoint was then publicly exposed using an Ngrok secure tunnel [15], enabling the Android application to seamlessly interact with the model via HTTP POST requests for real-time translation. This approach effectively addresses the limitations of on-device processing (storage, memory, and battery) inherent to complex neural network models.

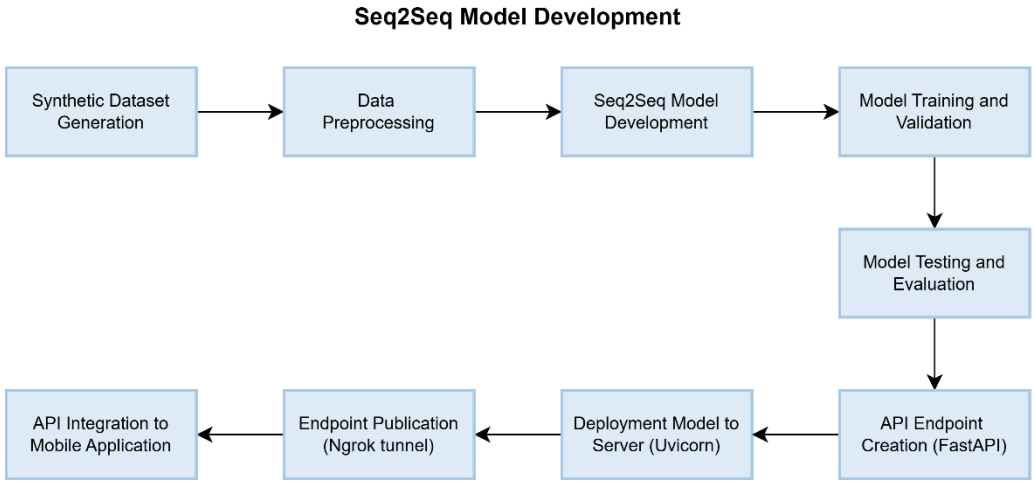

Fig. 4: Seq2Seq Model Deployment

The model inference flow, as depicted in Fig. 5, starts with the user's input sequence, which is immediately subjected to preprocessing to convert it into numerical tensors. This tensor then passes sequentially through the core model architecture, starting from the encoder block, decoder block, attention mechanism, concatenate layer, and finally to the output layer, which yields the final prediction. Once the generation process is complete, the

resulting token sequence undergoes post-processing to translate the numerical data back into a complete, readable sentence for the user.

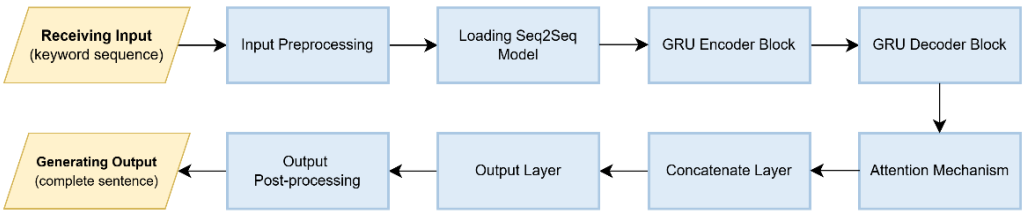

Fig. 5: Seq2Seq Model Inference within Mobile Application

5 Results, Analysis and Discussions

The final performance of the Seq2Seq translation model was assessed on a test set using 700 samples of keyword sequences, focusing on convergence stability and the dual requirements of AAC output quality (content fidelity and linguistic fluency).

Model training was configured for a maximum of 75 epochs but was terminated earlier at epoch 51 by the EarlyStopping callback. The optimal weights were restored from epoch 43, achieving the lowest validation loss of 1.22255 and the peak validation accuracy of 0.4543. Analysis of the training history confirms stable convergence. This stability is clear from the minimal separation between the training and validation loss curves (see Fig. 6), showing that the model generalized knowledge successfully and avoided major overfitting. This outcome validates the efficacy of regularization techniques, including label smoothing and gradient clipping.

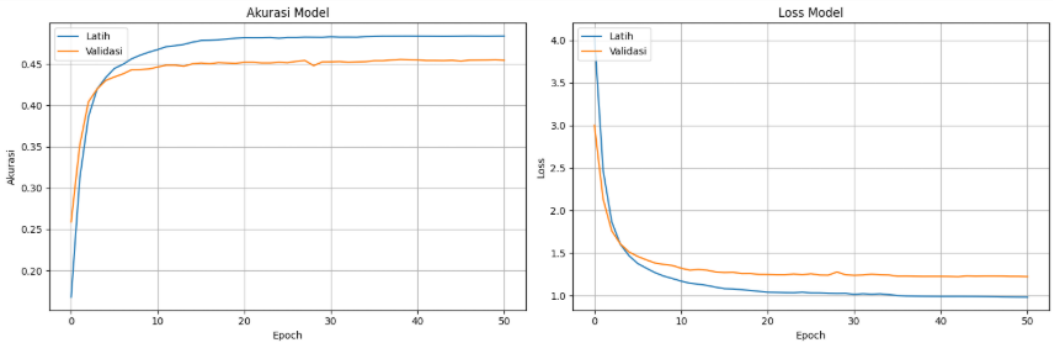

Fig. 6: Convergence Profile of the Seq2Seq Model during Training

5.1 Quantitative Results and Model Analysis

The model's translation quality was quantified using ROUGE and BLEU metrics. This dual-metric approach is necessary to ensure the output meets the semantic completeness and linguistic fluency. As shown in Table 1, the high ROUGE scores strongly validate that the model successfully preserves the core semantic content during the transformation, while the complementary BLEU score confirms the output sentences maintain the required linguistic fluency.

Table 1: Quantitative Evaluation Metrics on Test Set (ROUGE and BLEU)

Metric	Value	Justification for AAC Relevance
ROUGE-1 (F1)	94.94% (Good)	Indicates that most of the individual words from the reference sentence were successfully captured by the predicted sentence.
ROUGE-2 (F1)	87.12% (Good)	Indicates that most of the word pairs were successfully captured by the predicted sentence.
ROUGE-L (F1)	93.70% (Good)	Indicates that the model is very good at constructing sentences with a word sequence structure that is nearly identical to the reference sentence.
BLEU (4-gram)	42.95% (Good)	Places the model in the good category (benchmark 40–50), indicating that the model can produce output sentences that are similar to the reference sentence (high-quality translation).

5.2 Application Usability and Functional Validation

The application, Talk of the Heart AAC, was developed based on a modular architecture encompassing six integrated components (User Authentication, Profile Configuration, Parent Mode, Child Mode, Bidirectional Conversation, and Settings). The design addresses the unidirectional communication gap by focusing on two primary functions:

1. Speech Output: The Child Mode interface enables message composition through 155 picture cards which are color-coded following the Fitzgerald Key to support grammar learning. Speech generation is handled directly via the mobile device's native, offline Text-to-Speech (TTS) engine, allowing voice customization (type, pitch, speed) via the Settings module. Because the sound generation operates locally and does not rely on a cloud API, it requires zero internet bandwidth and does not add any network-related latency to the system's response time.
2. Bidirectional Messaging and Translation: This function is hosted within the Bidirectional Conversation module. It is the central feature where the Seq2Seq model endpoint receives the raw keyword sequence from the mobile client and translates it into a grammatically complete Indonesian sentence, facilitating asynchronous, real-time communication with the caregiver. The Parent Mode complements this by providing comprehensive control (CRUD operations) over custom cards and categories. To manage this application data efficiently, the system employs a hybrid database architecture. It uses Room [26] as a local database to store cards, categories, and user profiles, allowing for fast, offline access. This local data is seamlessly synchronized in real-time with Firebase Cloud Firestore (NoSQL) [27] for cross-device consistency, while Firebase Cloud Storage [28] handles larger media assets like profile photos and custom card illustrations.

The complete system workflow and core functional interfaces are visually documented in Fig. 7. This composite figure systematically illustrates the user workflow, including (a) Onboarding in Dark Mode, (b) Onboarding in Light Mode, (c) Avatar Selection, (d) Voice

Profile Configuration, (e) Card Creation in Parent Mode, (f) Picture Cards Composition in Child Mode, (g) Model Output in Conversation, and (h) Settings Menu.

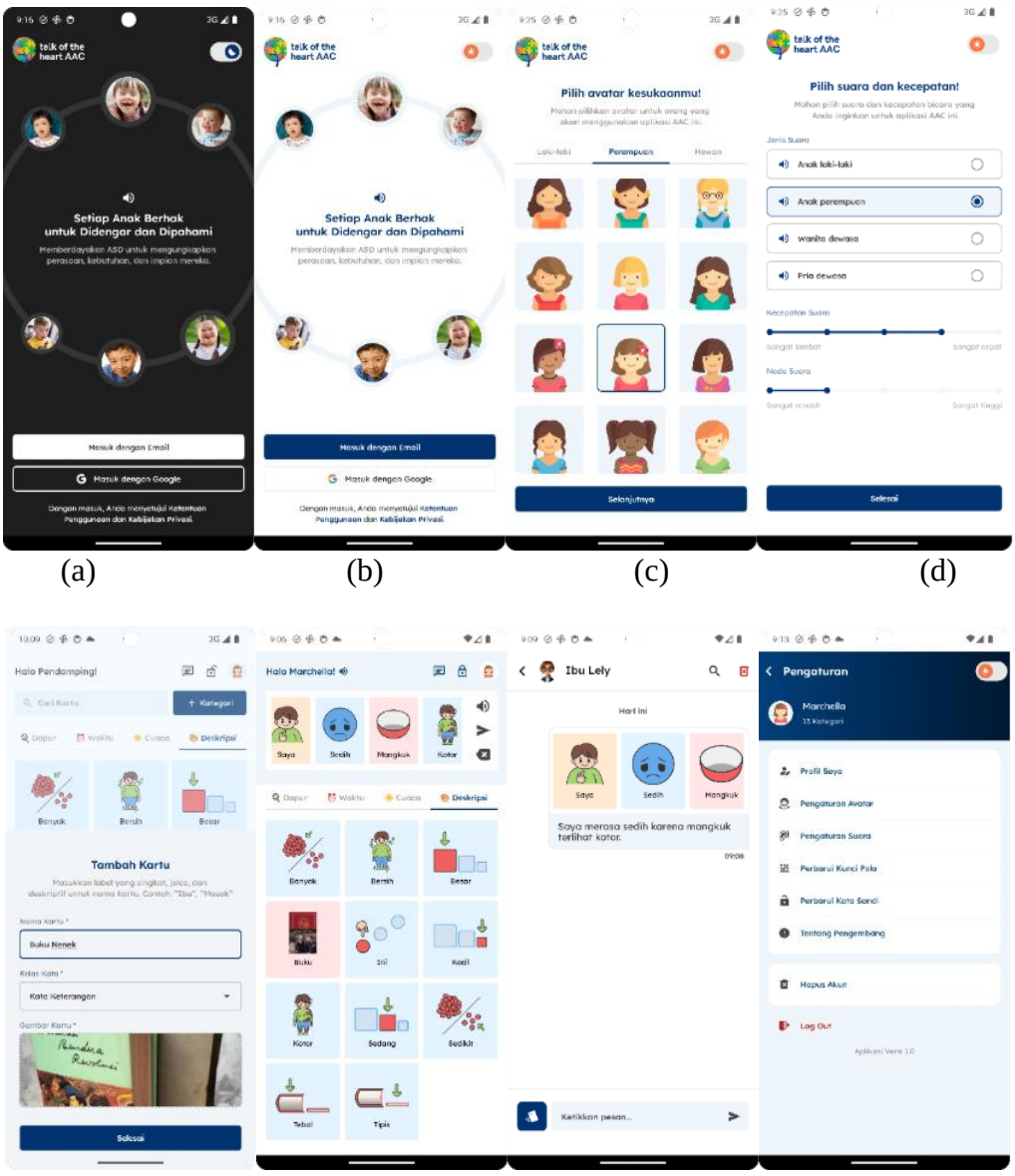

Fig. 7: User Interface of the Talk of the Heart AAC Application

System validation was performed using both functional and non-functional methods to confirm system reliability and user acceptance among the target demographic.

Functional validation via User Acceptance Testing (UAT), comprising 34 test scenarios across all application modules, achieved a 100% success rate. This result confirms that the system's architecture, including the translation API calls and supporting components, operates robustly according to the specified requirements. Usability was quantified via the System Usability Scale (SUS) [29], involving three distinct groups (each consisting of 2-3 autistic children and 1 teacher). The application achieved an average SUS score of 91.67. This score is categorized as "excellent" and exceeds the industry acceptability threshold (score ≥ 70), validating that the application is highly intuitive and user-friendly.

Qualitative feedback, systematically gathered through direct observations and personal discussions with teachers, provided critical insight:

1 Sensory Appropriateness: The visual design was assessed as highly appropriate for children with special needs. Reviewers consistently noted the clean, minimalist interface design, which effectively supports the child's focus and multi-sensory learning needs (visual, textual, auditory).
2 Audio Clarity and Customization: The TTS functionality, supporting fully customizable voice profiles (type, pitch, speed), was validated for its clarity and flexibility in addressing the child's varied sensory preferences.
3 Observed Interaction Behavior: Observations indicated that for children in the early stages of communication development, interaction often remained focused on vocabulary learning (tapping cards and repeating audio) rather than complex sentence construction, suggesting a need for features that differentiate based on developmental stage.
4 Operational Friction: Critical feedback highlighted operational issues, specifically system latency during resource-intensive tasks (e.g., adding new contacts and transmitting messages). To evaluate this, we measured the end-to-end response time on a Vivo Y33S device under a stable network (92.52 Mbps download, 77.80 Mbps upload, 6 ms ping). Processing takes 4 to 8 seconds depending on input length. For instance, 3 tokens take around 4 seconds, while 5 tokens take around 7 seconds. This delay is primarily caused by network transmission overhead from external API calls and the computational timesteps required by the model's autoregressive decoding.
5 Security Flaw: A vulnerability was identified in the pattern lock mechanism, where the security intended for Parent Mode was bypassed after simple observation, which degrades the effectiveness of separating the child and parent modes.
6 System Scalability Analysis: The application handles data storage efficiently because it uses Firebase, which naturally supports a growing number of users and media files. However, the translation API currently runs through an Ngrok tunnel as a Proof of Concept (PoC). This setup is only for testing and cannot process many concurrent user requests. To ensure fast and stable performance under heavy workloads, the API must be migrated to a dedicated cloud server with a load balancer.

6 Conclusion

We have successfully applied a Seq2Seq model for a bidirectional communication in AAC application called Talk of the Heart AAC. We summarize the analysis, modelling, system development and experimental results as follow:

1. The primary goal of implementing a linguistic model proficient in translating raw keyword compositions into complete, natural, and semantically accurate Indonesian sentences was achieved with demonstrably "good" performance. Translation quality was substantiated by a BLEU score of 42.95%, aligning the output within the "good" benchmark (40–50). Furthermore, the model's ability to preserve the core semantic content was confirmed by ROUGE-1 (94.94%), ROUGE-2 (87.12%), and ROUGE-L (93.70%) scores. However, the model exhibited a limitation in generalization failure when processing illogical input sequences that violate the structured patterns of the training data.
2. The secondary objective, which was to develop a functional and user-friendly Android application, was also fully achieved. Functional reliability was established

via UAT, which demonstrated a 100% success rate. Usability was quantified by the SUS, achieving an average score of 91.67. This result, categorized as "excellent," is directly supported by qualitative findings that validate the application's suitability for supporting the multi-sensory learning needs of autistic children.

Based on the operational and model constraints identified during the testing phase, the following recommendations are proposed for continued system enhancement:

1 Data Robustness: To address the model's generalization limitation, it is advised to increase the training corpus size (e.g., to 20,000 data pairs) and incorporate complex data variants to enhance processing capability.
2 Latency Mitigation: Operational performance requires enhancement to mitigate the reported time lag (latency) experienced during contact addition and message transmission.
3 Security Countermeasures: Implement a security mechanism to prevent the bypass of the pattern lock after simple observation, such as adding an automatic timeout or cooldown feature after a limited number of failed attempts.
4 Differentiated Learning: Future development should integrate a mode focused on vocabulary learning to support users in the initial stages of AAC use, accommodating varied communication developmental needs.

ACKNOWLEDGEMENTS

We especially thank Ibu Siti Asma, S.Pd., for her expert validation. Her professional check ensured that the dataset was correct and relevant for the communication needs of autistic children. We are also grateful to the participating teachers and children at the House of Faith learning center. Their practical testing and honest feedback were essential for making the final application easy and functional to use.

References

[1] M. G. Logrieco *et al.*, ‘Nonverbal Skills Evolution in Children with Autism Spectrum Disorder One Year Post-Diagnosis’, *Children*, vol. 11, no. 12, Dec. 2024, doi: 10.3390/children11121520.

[2] J. Damiao *et al.*, ‘Parent Perspectives on Assisted Communication and Autism Spectrum Disorder’, *American Journal of Occupational Therapy*, vol. 78, no. 1, Jan. 2024, doi: 10.5014/ajot.2024.050343.

[3] E. Lorang, N. Maltman, C. Venker, A. Eith, and A. Sterling, ‘Speech-language pathologists’ practices in augmentative and alternative communication during early intervention’, *AAC: Augmentative and Alternative Communication*, vol. 38, no. 1, pp. 41–52, 2022, doi: 10.1080/07434618.2022.2046853.

[4] N. M. Alzrayer, ‘Special Education Teachers’ Perspectives Toward Tablet-Based Augmentative Alternative Communication (AAC) Devices’, *International Education Studies*, vol. 17, no. 4, p. 51, Jul. 2024, doi: 10.5539/ies.v17n4p51.

[5] L. Hiryanto, M. Angelina, and O. Deliani Hutagaol, ‘UI/UX Prototype of Visually Interactive Communication and Reading Aid for Autistic Children with Speech Disability’. [Online]. Available: https://s.id/1Wd0R. [Accessed: Nov. 15, 2025].

[6] Saltillo Corporation, ‘TouchChat HD - AAC’, https://apps.apple.com/us/app/touchchat-hd-aac/id398860728. [Accessed: Nov. 18, 2025].

[7] Saltillo Corporation, ‘TouchChat: The gold standard in AAC solutions’, https://touchchatapp.com/. [Accessed: Nov. 18, 2025].

[8] A. K. Pandey and S. S. Roy, 'Natural Language Generation Using Sequential Models: A Survey', *Neural Process Lett*, vol. 55, no. 6, pp. 7709–7742, Dec. 2023, doi: 10.1007/s11063-023-11281-6.

[9] M. M. Henry, G. N. Elwirehardja, and B. Pardamean, 'Automatic question generation for bahasa indonesia examination using copynet', *Procedia Comput Sci*, vol. 245, pp. 953–962, 2024, doi: 10.1016/j.procs.2024.10.323.

[10] S. N. Damayanti, A. Prasetiadi, and A. R. Dewi, 'Automatic Story Text Generation Using Recurrent Neural Network Algorithm', in *2024 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT)*, IEEE, Nov. 2024, pp. 349–354. doi: 10.1109/COMNETSAT63286.2024.10861931.

[11] L. Hiryanto *et al.*, 'Multiple Themes in Augmentative and Alternative Communication Mobile Application for Autistic Children with Speech Difficulties', *Journal of Southwest Jiaotong University*, vol. 59, no. 2, 2024, doi: 10.35741/issn.0258-2724.59.2.10.

[12] K. Sun, T. Qian, X. Chen, and M. Zhong, 'Context-aware seq2seq translation model for sequential recommendation', *Inf Sci (N Y)*, vol. 581, pp. 60–72, Dec. 2021, doi: 10.1016/j.ins.2021.09.001.

[13] H.-I. Liu and W.-L. Chen, 'X-Transformer: A Machine Translation Model Enhanced by the Self-Attention Mechanism', *Applied Sciences*, vol. 12, no. 9, p. 4502, Apr. 2022, doi: 10.3390/app12094502.

[14] VICARA, 'VICARA 3 (Visually Interactive Communication and Reading Aid)', https://play.google.com/store/apps/details?id=com.vicaraxebp.vicara3&hl=id. [Accessed: Nov. 14, 2025].

[15] I. Sutskever, O. Vinyals, and Q. V. Le, 'Sequence to Sequence Learning with Neural Networks', Dec. 2014, [Online]. Available: http://arxiv.org/abs/1409.3215. [Accessed: Nov. 17, 2025].

[16] O.-C. Novac *et al.*, 'Analysis of the Application Efficiency of TensorFlow and PyTorch in Convolutional Neural Network', *Sensors*, vol. 22, no. 22, p. 8872, Nov. 2022, doi: 10.3390/s22228872.

[17] F. J. J. Joseph, S. Nonsiri, and A. Monsakul, 'Correction to: Keras and TensorFlow: A Hands-On Experience', 2021, pp. C1–C1. doi: 10.1007/978-3-030-66519-7_12.

[18] B. T. Chicho and A. Bibo Sallow, 'A Comprehensive Survey of Deep Learning Models Based on Keras Framework', *Journal of Soft Computing and Data Mining*, vol. 2, no. 2, Oct. 2021, doi: 10.30880/jscdm.2021.02.02.005.

[19] J. Bernhard, 'Alternatives to the Scaled Dot Product for Attention in the Transformer Neural Network Architecture', Nov. 2023, [Online]. Available: http://arxiv.org/abs/2311.09406. [Accessed: Nov. 19, 2025].

[20] S. Anuradha and M. Sheshikala, 'Investigating the recall efficiency in abstractive summarization: an experimental based comparative study', *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 39, no. 1, Jul. 2025, pp. 446-454, doi: 10.11591/ijeecs.v39.i1.

[21] C.-Y. Lin, 'ROUGE: A Package for Automatic Evaluation of Summaries'.

[22] L. Yang and S. Qiu, 'BLEU Function Analysis of Machine Translation Based on Transformer Model', in *Proceedings of the 2024 International Conference on Artificial Intelligence, Digital Media Technology and Interaction Design*, New York, NY, USA: ACM, Nov. 2024, pp. 230–236. doi: 10.1145/3726010.3726046.

[23] 'Evaluative Comparison of ASGI Web Servers: A Systematic Review', *International Journal of Science and Engineering Applications*, Mar. 2024, doi: 10.7753/ijsea1303.1007.

[24] Uvicorn Developers, 'Uvicorn: The lightning-fast ASGI server', https://uvicorn.dev/. [Accessed: Nov. 20, 2025].
[25] ngrok, 'Secure Tunnels - ngrok', https://ngrok.com/. [Accessed: Nov. 16, 2025].
[26] Google Developers, 'Save data in a local database using Room', 2024, [Online]. Available: https://developer.android.com/training/data-storage/room. [Accessed: Mar. 9, 2026].
[27] Google, 'Cloud Firestore Documentation', 2024, [Online]. Available: https://firebase.google.com/docs/firestore. [Accessed: Mar. 9, 2026].
[28] Google, 'Cloud Storage for Firebase Documentation', 2024, [Online]. Available: https://firebase.google.com/docs/firestore. [Accessed: Mar. 9, 2026].
[29] P. Vlachogianni and N. Tselios, 'Perceived usability evaluation of educational technology using the System Usability Scale (SUS): A systematic review', *Journal of Research on Technology in Education*, vol. 54, no. 3, pp. 392–409, May 2022, doi: 10.1080/15391523.2020.1867938.

Notes on contributors

Lely Hiryanto (Member, IEEE) received the Bachelor in Computer Science from Tarumanagara University, Indonesia in 2001, Postgraduate Diploma, Master of Science and PhD in Computer Science from Curtin University in 2015, 2016 and 2022, respectively. She is currently an associate professor with the Faculty of Information Technology, Tarumanagara University, Jakarta, Indonesia. Her research interests include Data Science, Mathematical Programming, and Network Optimization.

Marchella Angelina received the bachelor's degree in computer science from Tarumanagara University, Indonesia, in 2026. Her technical interests include Project Management, Product Design, and Mobile Development.

Tony Tony (Member, IEEE) received the bachelor's degree in computer science from Tarumanagara University, Indonesia, in 2005, the Master of Computer Science degree from the University of Indonesia, in 2010 and the Ph.D. degree from Curtin University, Perth, Australia, in 2021. He is currently a Senior Lecturer with the Faculty of Information Technology, Tarumanagara University, Indonesia. His research interests include wireless sensor networks, information systems, and database.