Jurnal Teknik Informatika Vol. 17 No. 2, Oktober 2024 (203-212)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v17i2.41240

JURNAL TEKNIK INFORMATIKA

Homepage : http://journal.uinjkt.ac.id/index.php/ti

Enhancing Speech-to-Text and Translation Capabilities for Developing Arabic Learning Games: Integration of Whisper OpenAI Model and Google API Translate

Dewi Khairani[1*], Tabah Rosyadi[2], Arini[3], Imam Luthfi Rahmatullah[4], Fauzan Farhan Antoro[5]

[1,2,3,4,5]Department of Informatics Engineering, Faculty of Science and Technology, State Islamic University Syarif Hidayatullah Jakarta
[1,2,3,4,5]Jl.Ir H. Juanda Street Number 95, Ciputat, West Tangerang, Banten, Indonesia
E-mail: [1]dewi.khairani@uinjkt.ac.id, [2]tabah.rosyadi@university.ac.id

Article:
Accepted: August 24, 2024
Revised: July 12, 2024
Issued: October 29, 2024

***Correspondence Address:**
dewi.khairani@uinjkt.ac.id

ABSTRACT

This study tackles language barriers in computer-mediated communication by developing an application that integrates OpenAI's Whisper ASR model and Google Translate machine translation to enable real-time, continuous speech transcription and translation and the processing of video and audio files. The application was developed using the Experimental method, incorporating standards for testing and evaluation. The integration expanded language coverage to 133 languages and improved translation accuracy. Efficiency was enhanced through the use of greedy parameters and the Faster Whisper model. Usability evaluations, based on questionnaires, revealed that the application is efficient, effective, and user-friendly, though minor issues in user satisfaction were noted. Overall, the Speech Translate application shows potential in facilitating transcription and translation for video content, especially for language learners and individuals with disabilities. Additionally, this study introduces an Arabic learning game incorporating an Artificial Neural Network using the CNN algorithm. Focusing on the "Speaking" skill, the game applies to voice and image extraction techniques, achieving a high accuracy rate of 95.52%. This game offers an engaging and interactive method for learning Arabic, a language often considered challenging. The incorporation of Artificial Neural Network technology enhances the effectiveness of the learning game, providing users with a unique and innovative language learning experience. By combining voice and image extraction techniques, the game offers a comprehensive approach to enjoyably improving Arabic speaking skills.

Keywords : *machine translation; speech to text; artificial neural network; openAI; whisper model;*

Jurnal Teknik Informatika Vol. 17 No. 2, Oktober 2024 (203-212)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v17i2.41240

1. INTRODUCTION

Language learning has always been an essential part of education, it enabling people to communicate effectively in different cultural contexts. In the advancement of technology, applications and games for educational purposes have emerged as effective tools to facilitate the language learning. In particular, Arabic learning games that gained significant popularity due to the growing interest in learning the Arabic language for various purposes, such as religious believes, cultural understanding and business opportunities [1].

Arabic is essential to learn for Muslims because as a Muslims we must study the Qur'an to understand and practice it well [2]. Being one of the world's most widely spoken languages, Arabic gives learners unique challenges. Its complex grammar, diverse dialects, and unique writing system make it a challenging language to master. Therefore, learning Arabic must be packaged as light, interesting, and fun. To make it happen, many media can be used in today's sophisticated era, one of which is by using smartphones [3].

Traditional language learning methods often fall short in providing an immersive and engaging experience that caters to the needs of modern learners. This is where technology can play a pivotal role in enhancing language learning experiences. Learning Arabic with games on mobile devices such as Android will be easier and more fun because it is more attractive with a user interface that makes users not feel bored. The concept of using game for learning is made so that users continue to be interested and enthusiastic [4]. The integration of advanced language processing technologies, such as speech-to-text and translation capabilities, can significantly enhance the effectiveness and user experience of Arabic learning games. These technologies can provide real-time feedback and personalized learning experiences, making the process more engaging and interactive for learners. Additionally, incorporating cultural elements and real-life scenarios into the games can further immerse users in the language and enhance their overall understanding and retention.

Speech-to-text and translation capabilities are crucial for enhancing the effectiveness of Arabic learning games These technologies enable learners to interact with the games using their voices, allowing for a more natural and immersive learning experience [5].. By speaking Arabic phrases and sentences, learners can practice pronunciation and improve their speaking skills in a realistic context.

Furthermore, translation capabilities allow learners to understand and analyze Arabic texts more effectively. Arabic is a language with complex grammar and syntax, and learners often struggle to understand sentences' meaning and structure [6]. By integrating translation services, learners can receive instant translations and explanations of Arabic texts, facilitating comprehension and improving their overall language proficiency.

Four kinds of skills in learning Arabic will be included in this Arabic learning game [7]. The four skills in question are listening (istima'), speaking (kalam), reading skills (qira'ah), and writing skills (kitabah). However, in developing this application, the author only related on developing one skill, namely speaking skills (kalam).

By incorporating text-to-speech and transcription features into the Arabic learning game, this study aims to explore the integration of OpenAI's Whisper model and Google API Translate in mobile applications. By harnessing the capabilities of these advanced technologies, developers can create immersive and effective Arabic learning games that cater to a diverse range of learners.

1.1. Overview of Whisper OpenAI Model

OpenAI created the Whisper OpenAI Model, which is an automatic speech recognition (ASR) system. The Whisper model developed by OpenAI is designed explicitly for normal speech recognition, not whispered speech recognition [8]. Whisper is an ASR system that utilizes deep learning techniques to convert spoken language into written text. It is trained on a vast amount of multilingual and multitask data, making it highly versatile and capable of performing well across various languages and domains. This is achieved by training the model on a diverse corpus of speech data, including examples from different dialects, accents, colloquialisms, and slangs. The model has an attention mechanism that allows it to focus on different parts of the input speech signal depending on what is being

Jurnal Teknik Informatika Vol. 17 No. 2, Oktober 2024 (203-212)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v17i2.41240

transcribed. Figure 1 shows Whisper's ability to compete with commercial and open-source state-of-the-art ASR systems in long-form transcription. These results show that in addition to being a robust model, Whisper's performance is also close to that of a human translator.

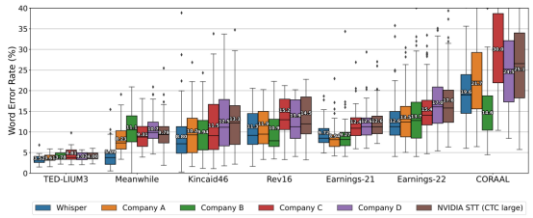

Figure 1. *Comparison of whisper with other ASR models on long form transcription [9]*

One of the key features of the Whisper ASR system is its ability to handle noisy and accented speech. It has been trained on diverse data, including speech from different speakers, various recording conditions, and different accents. This training data diversity allows Whisper to recognize and transcribe speech in challenging acoustic environments.

Integrating the Whisper ASR system into Arabic learning offers several benefits [10]. It enables real-time assessment and feedback on pronunciation and speech fluency, which are crucial aspects of language learning. By providing accurate and immediate feedback, Whisper can help learners improve their pronunciation and fluency more effectively.

The integration of Whisper into Arabic learning games enhances the interactive and immersive nature of the learning experience. Learners can engage in spoken interactions with the game, allowing for a more dynamic and engaging learning environment. This interactive approach can increase motivation and engagement, leading to better learning outcomes [11].

In addition, Whisper can support the development of personalized learning experiences. By analyzing the transcriptions generated by the ASR system, the game can adapt its content and difficulty level based on the learner's performance. This personalized approach ensures that learners receive tailored instruction, optimizing their learning progress.

Various studies have extensively evaluated the Whisper ASR system's accuracy and performance. Research conducted by OpenAI has demonstrated that Whisper achieves state-of-the-art performance on several benchmark datasets, including the LibriSpeech dataset [12]. It outperforms previous ASR systems in terms of word error rate (WER) and exhibits robustness to different acoustic conditions and accents.

However, it is worth noting that the accuracy of the Whisper ASR system may vary depending on the specific language and domain. While it performs exceptionally well in English and other widely spoken languages, its performance in certain low-resource languages may be relatively lower due to the limited availability of training data.

1.2. Introduction to Google API Translate

The integration of Whisper and Google API Translate has gained significant attention in recent years due to its potential in enhancing speech-to-text and translation capabilities. Whisper's integration with Google API Translate, a powerful machine translation service, allows leveraging both technologies for improved speech-to-text and translation capabilities [13]. The technical considerations for this integration involve ensuring compatibility between the two systems, handling data transfer, and addressing any language-specific challenges. It is requires careful preprocessing and alignment of the input data. This involves handling audio files, performing speech recognition using Whisper, and aligning the recognized text with the original audio. Additionally, the text output from Whisper needs to be formatted appropriately for translation using Google API Translate, considering factors such as sentence segmentation and language-specific preprocessing.

Google API Translate supports a wide range of languages, which presents opportunities and challenges for integration with Whisper. Language-specific considerations include handling language pairs, ensuring accurate translation quality, and addressing any limitations or biases in the translation output. Additionally, fine-tuning the integration for specific languages can further enhance the overall performance and user experience.

The integration of Whisper and Google API Translate holds significant potential in enhancing speech-to-text and translation capabilities which is one of the concerns of this research.

In term of the Arabic Automatic Speech Recognition (ASR) initially it involved the utilisation of recurrent neural networks along with Connectionist Temporal Classification (CTC) [14]; and this end-to-end deep learning models have greatly improved automatic speech recognition (ASR) performance by enabling direct learning from the audio waveform, eliminating the need for intermediate feature extraction layers [15]. An obstacle faced by end-to-end ASR models is the significant need for annotated data, especially for languages with limited resources like different forms of Arabic. In response to this issue, there is a growing interest in the use of self-supervised and semi-supervised learning methods. Models like Wav2vec2.0 and XLS-R first acquire valuable representations from extensive volumes of unlabeled or partially labelled data. These models can then be fine-tuned for particular tasks [16].

In developing the application in this study, the author will use Python to create a desktop-based application. The author utilizes the Pytorch framework using the Python language with the help of the Whisper library owned by OpenAI which is open source to perform transcription. PyTorch is a machine learning library for Python that focuses on usability and speed. PyTorch provides a Pythonic programming style that makes debugging easy and consistent with other popular scientific computing libraries, while remaining efficient and supporting GPU hardware accelerators[17]. In using the Whisper model, this library is needed to access the Whisper model in python.

Voice input can be of three types: microphone sound, computer system speaker sound, and video or audio files. This input is then processed and the results will be displayed on the screen, the results can be selected in the form of transcription only, transcription that has been translated into the language of choice, or a combination of both. To translate the transcription results into another language, the author will utilize the API and the deep_translator library to use the Google Translate service.

2. METHODS

A software development life cycle describes the manufacturing procedure in developing an application or the model. In the first research step, we developed a machine learning and deep learning model consisting of six stages: Data Collection, Data Preprocessing, Model Training, Model Evaluation, and Model Serving / Development.

The primary goal of this research is to develop Arabic learning games that use the Whisper OpenAI model for speech-to-text conversion and Google API Translate for accurate translation, improving learners' pronunciation and comprehension of Arabic. [18].

After that, Arabic speech samples will be collected from native and non-native speakers for training and testing purposes. These samples will include common phrases, vocabulary, and sentences relevant to language learning. The author uses a dataset that is very suitable for the author's needs. The dataset is called the Arabic Speech Commands Dataset created by Abdulkader Ghandoura in 2021, published on the Zenodo website (https://zenodo.org/record/4662481#.YkVMqZlBxEY). This dataset has the following characteristics:

a. The dataset has 40 folders (classes) of Arabic words, each spoken by 30 different people. Each person repeats the pronunciation 10 times.
b. Each folder (class) consists of 300 files in WAV (Waveform Audio File) format.
c. The total of all existing files is 30 people × 10 repetitions × 40 classes = 12000 WAV files.
d. Each WAV audio is exactly 1 second long.
e. The dataset has a sampling rate of 16000 Hz.
f. 16-bit little-endian signed PCM modulation format.
g. 256 kbps bitrate, mono audio channel (1 channel).

The Arabic speech data will undergo preprocessing, including noise reduction, segmentation into phrases or sentences, and conversion into a format compatible with the Whisper model. A training set comprising labeled Arabic audio clips paired with

corresponding transcriptions will be created. This dataset will also include non-native pronunciations for robustness. We also create a function to calculate the duration distribution of the audio dataset and visualize it into a histogram, it can be seen that the entire audio is 1 second long so that there are no outliers and imbalances in this dataset.

The Whisper OpenAI model will be fine-tuned using the collected Arabic speech data. This will enhance its ability to recognize specific Arabic phonemes, accents, and dialects. Hyperparameters such as learning rate and batch size will be adjusted for optimal performance. The Google API Translate will be integrated into the system to translate recognized speech in real-time. API calls will be configured to ensure seamless interaction between the game and the translation engine.

For Model Evaluation in Speech Recognition, the performance will be evaluated based on its Word Error Rate (WER) [19], focusing on the accuracy of recognizing Arabic phrases and words, especially for non-native learners. Language learners will test the speech-to-text and translation capabilities in a game environment. Feedback on usability, accuracy, and response times will be gathered.

We modified specific parameters of the Whisper model to optimize the performance of Arabic learning game we previously developed [20]. These modifications primarily focused on decoding adjustments to balance speed and accuracy. First, we adjusted the suppress token parameter by modifying its value allowing more flexible token handling during transcription. Second, we increased the no-speech threshold to improve the model's ability to detect speech amidst silence. Lastly, we created two presets for sampling methods: greedy and beam search. The greedy preset was optimized for speed by setting the temperature parameter to 0 and reducing both the best_of and beam_size parameters from 5 to 0. In contrast, the beam search preset aimed to enhance accuracy by adjusting the best_of and beam_size parameters. These adjustments were designed to balance processing speed and transcription accuracy, with the beam search preset offering more precise results while the greedy preset sped up processing.

Once the models have been optimized, they will be fully integrated into the Arabic learning game platform. The game will be deployed on both mobile and web platforms to maximize accessibility for users. The Whisper model and Google API will be deployed cloud-based to support real-time speech recognition and translation capabilities. This setup will ensure scalability and maintain efficient processing times.

3. RESULTS AND DISCUSSION

In this section, we discussed the model developed that aimed at classifying the sound classes present in the dataset. A predictive modeling approach is employed, focusing on determining which class the incoming data most closely aligns with. Instead of using fine-tuned or pre-trained models, the author opts for a custom approach due to the mismatch between the available models and the specific structure of the dataset. To address this, a 1-dimensional Convolutional Neural Network (CNN) is implemented, tailored to fit the unique characteristics of the dataset.

The results of the data training are made in a useful visualization to make it easier to read the data results. The Figure 2 are the plotting results obtained by the author after successfully conducting data training with the model created.

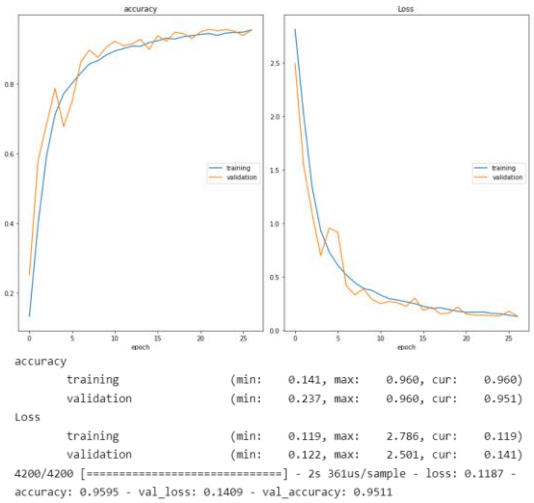

Figure 2. *Accuracy and loss visualization results*

Figure 2 shows the results using the Adam optimizer. The model achieved satisfactory performance, with an accuracy of 96% in both training and validation data. Subsequently, the model was saved in the h5 format for future use. The complete training result is shown in Table 1. It is evident that most classes are accurately predicted, as indicated by the darker shading on the tested labels. For

instance, the "cancel" label (in Arabic: لغاء or 'iilgha') is correctly predicted in most cases.

The author tested the model using their own voice recorded through a Python program to further evaluate it. After recording, the trained model was called upon to make predictions. The model was then tested by calling the API deployed on Heroku, utilizing the Postman application for REST API testing. The following are the results of these tests:

Table 1. *Training result*

Optimizer	accuracy	loss	val_ac	val_loss
Adam	**0,9674**	**0,0948**	**0,9667**	**0.1123**

In using the model, post-processing is performed by filtering the obtained results. The filter for this post-processing is obtained by noting hallucinations that may occur during the recording process. Additionally, efforts are made to obtain optimal results by modifying parameters in model usage. This iterative process allows developers to fine-tune the model and improve its performance over time. By continuously refining the model through post-processing and parameter adjustments, developers can ensure that it remains accurate and reliable in real-world applications.

The parameters modified in model usage are related to decoding, as follows:

a. Supress Token. Using the suppress token parameter is changed from -1 to empty.
b. No Speech Threshold. The no speech threshold parameter is changed from 0.6 to 0.72.
c. Sampling-related Parameters. For sampling-related parameters, 2 presets are created that can be used for greedy and beam search sampling. Greedy can speed up the process, while beam search makes the process more accurate. The details are as follows:
    1. Greedy: changing the temperature parameter to 0, and changing the best_of and beam_size parameters from 5 to 0.
    2. Beam search: changing the best_of and beam_size parameters from 5 to 3 and the patience parameter from empty to 1. The value 3 for the best_of and beam_size parameters is chosen as the middle value between accuracy and speed.

The construction phase outlines the product development process and is the core stage where the author invests the majority of their time and effort into developing the application. During this phase, the author writes code and implements the functionalities meticulously designed in the previous stage, striving to achieve high-quality results and eliminate bugs.

Next in incorporating recording logic the program will give the user the option to use input thresholding or not. If the user chooses not to use a threshold, the program will record all incoming inputs, including during the recording session. If using a threshold, the user can choose to set their own limit or use the VAD available in the application, namely WEBRTC VAD and Silero VAD. The use of this threshold can help filter out noise, resulting in more meaningful results.

The transition phase represents the final stage in the development process, where the application is prepared for public release. This crucial phase involves getting the application ready for deployment and ensuring it is set for user interaction. Before officially launching the application, it must undergo thorough functionality testing to confirm that it meets the specified requirements and quality standards.

This testing process includes both white box and black box methods to comprehensively evaluate the application's performance and reliability. The development process will be revisited if any deficiencies are discovered during this testing—whether they pertain to potential errors, performance inefficiencies, or issues in the code implementation. In such cases, the application will be refined and improved based on the feedback from the testing phase, ensuring that all identified problems are addressed before the final release.

The filtering of results performed by the application has successfully overcome the hallucinations generated by the model. This filter is particularly useful, especially for the live speech recording feature in the application. It can be seen in Figure 3 that the use of the filter makes the results consistent with the input provided. The filter used successfully eliminates hallucinatory words generated by the model. The use of this filter can help make the results more consistent, especially when using the application in noisy environments.

Jurnal Teknik Informatika Vol. 17 No. 2, Oktober 2024 (203-212)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v17i2.41240

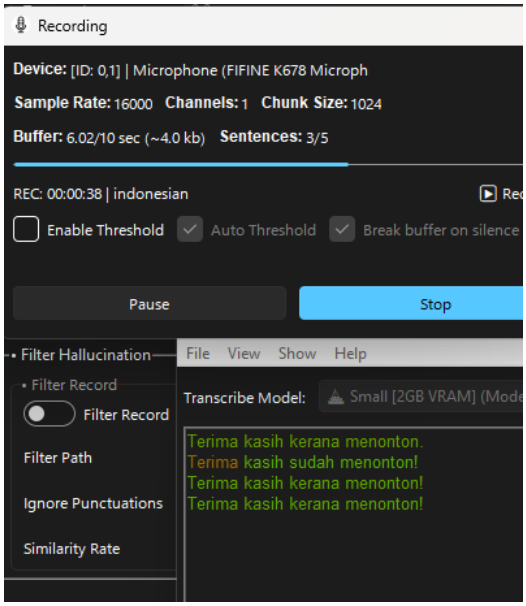

Figure 3. *Results when the filter is turned off*

Efforts to evaluate the performance and accuracy of the Whisper model and Google Translate machine translation involved a comprehensive analysis of translation accuracy using human speaker. This analysis aimed to assess how well the models performed in practical scenarios. The findings revealed that the implementation of the Faster Whisper model resulted in a significant increase in the application's efficiency while preserving the high accuracy levels achieved. This efficiency improvement was particularly noticeable when using the live speech recording feature, where the delay experienced by users became more apparent with different model configurations.

In evaluating the accuracy, it was found that Google Translate delivered more accurate outcomes than the Whisper model. This result aligns with expectations, given that Whisper excels in transcription but is less effective in translation [21]. As shown in Figure 13, the Whisper model struggles with translation tasks, especially when using its tiny and base variants, which fail to produce satisfactory translation results.

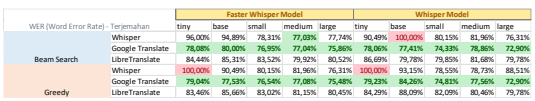

Figure 4. *Accuracy comparison*

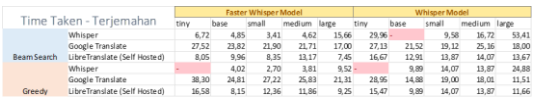

Figure 5. *Time comparison*

The Word Error Rate (WER) values for all translations exceed 70%, but this high WER does not necessarily indicate that the translation results are inaccurate. This is because translation can involve multiple valid language options for a given word. Additionally, WER calculations are limited in their applicability to machine translation, as they focus solely on surface-level word errors without accounting for the contextual and syntactic roles of words—factors that are crucial for assessing translation quality.

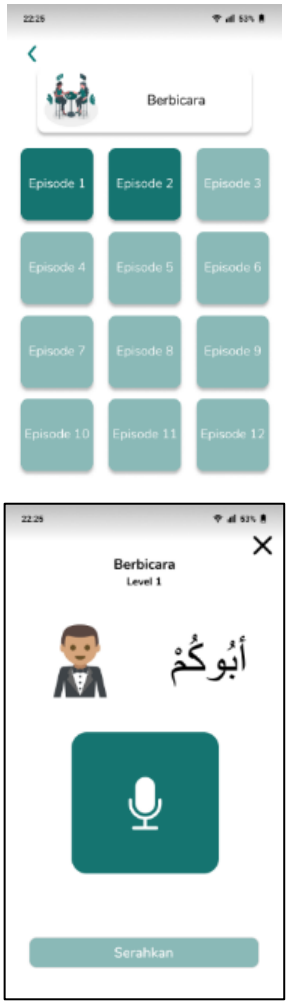

Figure 6. *The arabic game developed*

Figure 6 display the Arabic Game Learning we developed. We will conduct a questionnaire to measure the application's use. The success of a quasi-experimental study can be measured by several factors, such as the progress in students' Arabic language skills. This progress can be assessed in an experiment through tests or examinations conducted before and after using the Arabic language game. If the test results after using the game show a

significant improvement compared to the pre-game tests, it indicates that the quasi-experimental results are successful. Another factor is student satisfaction, which can be measured after the use of the Arabic language game.

If students feel more engaged and find it easier to understand Arabic after using the game, this also suggests successful quasi-experimental outcomes. Additionally, an important factor is the improvement in students' Arabic language skills, which can be observed through their ability to speak, write, and read Arabic after using the game. A significant improvement in these skills after using the game would indicate successful results as shown in Table 2.

Table 2. *Post-test result*

	1	2	3	4	5	6	7	8	9	10
EXPERIMENT GROUP	8.18	8.95	8.80	7.94	7.95	7.90	9.23	8.80	8.91	7.62
CONTROL GROUP	7.82	8.61	8.42	7.78	7.71	7.62	8.64	8.01	8.61	7.19

After testing and analyzing the data, the author found that the experimental group significantly improved Arabic language proficiency compared to the control group. This is evident from the Arabic language test results conducted before and after using the Arabic learning game. The experimental group also demonstrated higher engagement in learning Arabic and greater satisfaction with the game.

This suggests that learning games can be an effective alternative for enhancing language skills, especially in Arabic learning, which is often considered challenging. However, the study also observed that the gap between the two experimental groups remains small, suggesting that other factors beyond the use of the Arabic learning game may also influence students' language abilities. Therefore, further research is needed to identify these factors and ensure that the use of the Arabic learning game truly enhances students' Arabic language skills.

Additionally, more extensive and comprehensive research is required to develop and refine the Arabic learning game to impact Arabic language learning significantly. The developed game should address challenges in Arabic learning, such as motivating students to engage in more active and structured Arabic language study.

CONCLUSION

In conclusion, the study highlights the successful integration of advanced technologies to enhance Arabic language learning through a comprehensive application. The developed application, deployed on the Heroku cloud platform, utilizes a Convolutional Neural Network (CNN) with an Adam activation function to detect 20 Arabic words. With a dataset comprising 6,000 samples across 20 classes—300 samples per class—the model achieved an impressive accuracy of 95.52% and a validation accuracy of 95.50%. This high level of accuracy, confirmed through rigorous testing with the author's own voice, underscores the model's reliability and effectiveness.

The application's deployment coupled with its integration with CNN, facilitated seamless updates and maintenance. Once the application was successfully integrated and deployed, the Android application was able to leverage the Cloud-hosted API for its functionalities, streamlining the implementation process on mobile platforms.

Additionally, the integration of the Whisper model with the Google Translate API has significantly enhanced the application's capabilities. Google Translate expanded the Whisper model's functionality to support translations into 133 languages, compared to its previous limitation of translating only to English. This integration also improved translation accuracy by reducing the Word Error Rate (WER) during transcription.

The use of the Stable Whisper library and post-processing filter has effectively mitigated hallucinations, resulting in more stable and accurate outputs. Meanwhile, the Faster Whisper library has optimized application efficiency, although greedy parameters slightly impacted accuracy.

Experimental post-test questionnaire results suggest that the application is efficient and effective for speech-to-text and translation tasks, though user satisfaction appears limited, potentially due to the small sample size of respondents. Despite this, the application's effectiveness and efficiency are well-supported. Future research with a larger sample size could offer deeper insights into user satisfaction and guide further enhancements.

Overall, the integration of Whisper and Google Translate, combined with a robust

sound detection model and efficient deployment on Heroku, represents a significant advancement in language learning technology. This comprehensive approach not only improves the accuracy and functionality of the application but also enhances the overall learning experience, paving the way for further innovations in Arabic language education.

Among the enhancements discussed in this paper, future work can be done to improve speech-to-text capabilities for this application further. This will involve fine-tuning the Whisper model to have high recognition accuracy of Arabic speech, especially for noisy speech and/or different accents and dialects. Other fruitful ways of improving the real-time transcription speed and accuracy involve advanced decoding strategies, such as dynamic beam search or hybrid models. Also, the integration of continuous learning mechanisms that will permit the model to, over time, continuously improve thanks to user interactions and feedback could lead to finer transcriptions. The second significant development in application improvement would aim at increasing efficiency in handling more extended audio or video content, possibly through fragmentation and immediate data processing, with smooth transcription of extended learning sessions. Lastly, the development should be directed toward increasing the scope of various file formats the application can support while improving accuracy in the transcription process of diverse media inputs, such as podcasts, lectures, or casual conversations, making the application even more functional within language learning contexts.

REFERENCES

[1] M. Rezi, J. Quintana, W. Dominic, and L. Darius, "Development of Educandy Platform as an Educational Game to Improve Arabic Language Learning Achievement," *Journal International Inspire Education Technology*, 2023, doi: 10.55849/jiiet.v2i1.445.

[2] I. S. Wekke, "Arabic Teaching and Learning: A Model from Indonesian Muslim Minority," *Procedia - Social and Behavioral Sciences*, 2015, doi: 10.1016/j.sbspro.2015.04.236.

[3] M. T. A. Ghani, M. Hamzah, W. A. A. W. Daud, and T. R. M. Romli, "The Impact of Mobile Digital Game in Learning Arabic Language at Tertiary Level," *Contemporary Educational Technology*, 2022, doi: 10.30935/cedtech/11480.

[4] H. Liang Ni, E. Fadzrin, and A. Shaubari, "Development of Think & Go Road Safety Mobile Game using Gamification Approach," *Applied Information Technology And Computer Science*, 2021.

[5] K. Chemnad and A. Othman, "Advancements in Arabic Text-to-Speech Systems: A 22-Year Literature Review," *IEEE Access*, 2023, doi: 10.1109/ACCESS.2023.3260844.

[6] K. F. Shaalan, "Arabic GramCheck: A grammar checker for Arabic," *Software - Practice and Experience*, 2005, doi: 10.1002/spe.653.

[7] D. Khairani, M. Iqbal, D. Rosyada, Z. Zulkifli, and F. Mintarsih, "Penerimaan Sistem Pembelajaran Bahasa Arab Dengan E-Learning dan Gim di Masa Pandemi COVID-19," *EDUKASI: Jurnal Penelitian Pendidikan Agama dan Keagamaan*, 2021, doi: 10.32729/edukasi.v19i3.958.

[8] Y. Peng *et al.*, "Reproducing Whisper-Style Training Using An Open-Source Toolkit And Publicly Available Data," 2023, doi: 10.1109/ASRU57964.2023.10389676.

[9] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust Speech Recognition via Large-Scale Weak Supervision," 2023.

[10] A. Waheed, B. Talafha, P. Sullivan, A. R. Elmadany, and M. Abdul-Mageed, "A Robust Dialect-Aware Arabic Speech Recognition System," 2023.

[11] M. M. Duisenova and A. N. Zhorabekova, "The effectiveness of gamification and artificial intelligence in increasing the motivation and effectiveness of students in learning English in elementary school," *Eurasia Journal of Mathematics, Science and Technology Education*, 2023, doi: 10.29333/ejmste/13670.

[12] R. Ma, M. Qian, M. J. F. Gales, and K. M. Knill, "Adapting an Unadaptable

ASR System," 2023, doi: 10.21437/Interspeech.2023-1899.
[13] V. R. and I. A. Funcke, "aiLangu - Real-time Transcription and Translation to Reduce Language Barriers," KTH Royal Institute of Technology, 2023.
[14] A. Ahmed, Y. Hifny, K. Shaalan, and S. Toral, "End-to-End Lexicon Free Arabic Speech Recognition Using Recurrent Neural Networks," in *Computational Linguistics, Speech and Image Processing for Arabic Language*, 2018.
[15] D. Wang, X. Wang, and S. Lv, "An overview of end-to-end automatic speech recognition," *Symmetry*. 2019, doi: 10.3390/sym11081018.
[16] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," 2020.
[17] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," 2019.
[18] S. Rangineni, "An Analysis of Data Quality Requirements for Machine Learning Development Pipelines Frameworks," *International Journal of Computer Trends and Technology*, 2023, doi: 10.14445/22312803/ijctt-v71i8p103.
[19] R. Yakubovskyi and Y. Morozov, "Speech Models Training Technologies Comparison Using Word Error Rate," *Advances in Cyber-Physical Systems*, 2023, doi: 10.23939/acps2023.01.074.
[20] R. Putri Fajriati, D. Khairani, N. Faizah Rozy, N. Husin, L. Wiyartanti, and T. Rosyadi, "Towards the Implementation of Arabic Language Mobile Apps Learning: Designed by User Insight," 2020, doi: 10.1109/CITSM50537.2020.9268901.
[21] D. Macháček, R. Dabre, and O. Bojar, "Turning Whisper into Real-Time Transcription System," 2024, doi: 10.18653/v1/2023.ijcnlp-demo.3.