A survey of automatic speech recognition deep models performance for Polish medical terms

Marta Zielonka[a;*], Wiktor Krasiński[a], Jakub Nowak[a], Przemysław Rośleń[a], Jan Stopiński[a], Mateusz Żak[a], Franciszek Górski[a] and Andrzej Czyżewski[a]

[a] Multimedia Systems Department, Faculty of Electronics, Telecommunications and Informatics at Gdańsk University of Technology
ORCiD ID: Marta Zielonka https://orcid.org/0000-0003-1407-6770,
Franciszek Górski https://orcid.org/0000-0001-7537-0039,
Andrzej Czyżewski https://orcid.org/0000-0001-9159-8658

Abstract. Among the numerous applications of speech-to-text technology is the support of documentation created by medical personnel. There are many available speech recognition systems for doctors. Their effectiveness in languages such as Polish should be verified. In connection with our project in this field, we decided to check how well the popular speech recognition systems work, employing models trained for the general Polish language. For this purpose, we selected 100 words from the International Classification of Diseases dictionary, the Polish-language version of the International Statistical Classification of Diseases and Health Problems. The words were read into a microphone by five women and five men and also generated with a speech synthesizer using a male and a female voice. This resulted in 1,200 recordings tested with the following systems: Whisper, Google speech-to-text, and Microsoft Azure speech-to-text. The achieved word recognition performance is reflected by the calculated metrics: *WER*, *WIL*, *Levenshtein distance*, *Jaccard distance*, *MER*, and *CER*. Results show that the highest efficiency for most cases was obtained by Azure speech-to-text. However, none of the tested models is ready for voice-filling medical records, describing cases, or prescribing treatment, because the number of errors made when converting speech to text is too high.

1 Introduction

Over the past decade, speech-to-text (STT) systems, also called automatic speech recognition (ASR) systems, have rapidly developed. The development was made possible by advances in deep learning theory and the growing demand for speech transcription systems or intelligent voice assistants. It included growth in the accuracy of these systems and an application for more languages than just English. Nowadays, the dominant approach to developing speech-to-text systems is based on neural networks, which achieve outstanding results in the transcription of recorded text. Systems such as DeepSpeech [11] or Recurrent Neural Network Transducer [5] can be given as examples. However, solutions using the Transformer-type architecture introduced in the [19] have received the most attention in recent years. Based on this idea, solutions such as OpenAI Whisper [16] and Conformer [10] have been developed that have achieved a Word Error Rate (*WER*) of less than 5% on different English datasets. Moreover, the authors in [16] show that their model can work for languages other than just English. As a result, high accuracy has been achieved in transcribing everyday speech, but there are still solutions tailored to some domains of applications that are insufficient. To address this issue in our work, we tested 3 systems: OpenAI Whisper, Google Speech-To-Text [3], and Azure Speech-To-Text [4] on 100 words from the International Classification of Diseases (ICD) dictionary for the Polish language. A list of medical terms derived from the dictionary is included for interested readers as supplementary material. Our team made recordings of these 100 words for 5 male voices, 5 female voices, 1 synthetic male, and 1 synthetic female voice, giving us 1200 recordings. We evaluated each STT system on this dataset and, for each of them, calculated a set of 6 different metrics designed for automatic speech recognition tasks. These metrics are *WER*, *WIL*, *Levenshtein distance*, *Jaccard distance*, *MER*, and *CER*. These metrics reflect how tested models are ready for voice-filling medical records, describing cases, or prescribing treatment, since the number of errors made when converting speech to text is crucial in this application domain.

2 Methods

This section describes methods; it includes a description of off-the-shelf, most common speech-to-text (STT) engines followed by metrics used in the experiments. The first subsection is dedicated to STT tools description, in which Whisper, Google STT, and Azure STT are included. The next subsection presents the metrics used, a description, and corresponding mathematical formulas.

2.1 Speech-to-text tools

There are various ready-to-use speech-to-text engines. Many of them support multiple languages. It is easy to notice that they work well for English on data that does not contain specialized domain terms. This study tests three of the most well-known tools with speech excerpts pronounced in Polish using medical terminology only. All these tools were used in a configuration that supports the Polish language; in other words, their authors trained them on datasets also in this language. These engines are Whisper (small, medium, large versions),

* Corresponding Author. Email: martaz@multimed.org

Google speech-to-text, and Azure speech-to-text, and those will be described briefly in the following subsections.

2.1.1 *OpenAI Whisper*

Whisper is an automatic speech recognition (ASR) system created by OpenAI. It was trained on 600,000 hours of multilingual data that was collected from the internet. Besides performing ASR, the system can translate speech into English upon language identification. OpenAI provides five model sizes: tiny, base, small, medium, and large. These variants vary in a number of parameters, which implies the difference in required VRAM (Video Random Access Memory) and speed. The system is constantly being improved; the most recent version (v20230314) was released on March 15th, 2023 [15]. Input audio for training was split into 30-second samples and converted into log-Mel spectrograms. For normalization, authors scaled input to the range from -1 to 1. The license for Whisper is granted by MIT.

2.1.2 *Google speech-to-text*

Google speech-to-text is an ASR service hosted in Google Cloud. The tool supports over 125 different languages and dialects, including Polish. There are 8 models to be used depending on the use case and data type we have - Polish is supported by four of them. Details about the algorithms used behind them are confidential. If needed, many Google speech-to-text models can be fine-tuned (the Polish language is supported for model adaptation). The service last update took place on the 7th of February, 2023. One can access Google's speech-to-text tool via a free home page preview or Application Programming Interface (API). The second approach is priced based on the number of speech excerpts successfully processed each month [9].

2.1.3 *Azure speech-to-text*

Azure speech-to-text is an automatic speech recognition service available in Azure cognitive services [4]. It can be used to transcribe speech in real-time or from recorded audio. Currently, it supports 141 languages and dialects. The exact model used is unknown. There is an option to fine-tune the base model with custom data for more specific use cases. Accessing the service can be done via API. The last update for speech-to-text service took place in February of 2023.

2.2 *Metrics*

The accuracy of speech-to-text machine learning models is often measured using a variety of metrics, including *WER*, *WIL*, *MER*, *Levenshtein distance*, *CER*, and *Jaccard distance*. Metric values are usually presented as fractions or percentages. A fraction of 0.5, for example, means that 50% of the results are not positive. These metrics assess the quality of the model's transcription output by comparing it to the ground truth or the original speech input. Following subsections provide an overview of these metrics and their applications in evaluating the performance of speech-to-text models, highlighting their strengths and limitations. All of the metrics were calculated using Python packages distance [1], which provides methods for calculating the similarity between arbitrary sequences, and jiwer [2], which is dedicated to the evaluation of ASR systems.

2.2.1 WER

Word Error Rate (*WER*) is one of the most popular metrics used to measure the quality of speech-to-text models [6], [20], [12]. It concentrates on the rate of words incorrectly transcribed in the output compared to the original input length. Precisely it sums: the number of word substitutions, number of word deletions, and number of word insertions, then divides this sum by the total number of words in the reference text. The formula to calculate the Word Error Rate (*WER*) is:

$$WER = \frac{S + I + D}{N}$$

where:

S - number of word substitutions (words that were recognized incorrectly and replaced with another word in the output)

I - number of word insertions (extra words in the output that were not present in the reference text)

D - number of word deletions (words in the reference text that are missing in the output)

N - total number of words in the reference text

2.2.2 WIL

Word Information Lost (*WIL*) rate provides a simple performance measure that varies from 0 when there are no errors to 1 when no hits. *WIL* indicates the percentage of incorrectly predicted words between ground-truth and hypothesis sentences. It is more suitable than *WER* for evaluating any application in which the proportion of word information communicated is more meaningful than the edit cost. At low error, both provide similar scores, so the inappropriate theoretical basis for the *WER* measure is not noticeable. [14] The formula to calculate the Word Information Lost (*WIL*) is the following:

$$WIL = 1 - \frac{H^2}{(H + S + D)(H + S + I)}$$

where:

S - number of word substitutions

D - number of word deletions

I - number of word insertions

H- number of hits (correctly transcribed words)

2.2.3 MER

Match Error Rate (*MER*), like *WIL*, provides a simple performance measure that varies from 0 when there are no errors to 1 when there are no hits. However, it has an even more intuitively simple probabilistic interpretation than *WIL*. *MER* is the percentage of words incorrectly predicted. Its value can be measured by subtracting from 1 the ratio of correctly transcribed words to the total number of words in the reference text. The formula to calculate the Match Error Rate (*MER*) is:

$$MER = \frac{S + D + I}{S + D + I + H} = 1 - \frac{H}{N}$$

where:

S - number of word substitutions

D - number of word deletions

I - number of word insertions

H- number of hits (correctly transcribed words)

N - total number of words in the reference text

2.2.4 Levenshtein distance

Levenshtein distance, also known as edit distance, is a metric used to quantify the difference between two strings of characters. It measures the minimum number of single-character edits (insertions, deletions, or substitutions) required to transform one string into another. It is widely used and considered a well-established string similarity measure, with many applications in natural language processing. Moreover, its great advantage is intuitiveness and ease of understanding. Its disadvantage, which is worth being aware of when using this metric, is that it is sensitive to changes in individual characters and can give high scores for small changes in long strings, even if most of the characters are the same.

2.2.5 CER

Character Error Rate (*CER*) is a metric that concentrates on the minimum number of character-level operations required to transform the reference text into the output text, similar to the *Levenshtein distance*. This metric is also commonly used in ASR task [8], [7], [18]. The formula to calculate the Character Error Rate (*CER*) is:

$$CER = S + I + DN$$

where:

S - number of character substitutions

D - number of character deletions

I - number of character insertions

N - total number of characters in the reference text

The *CER* score represents the percentage of characters in the reference text that were incorrectly predicted in the speech-to-text model output. The lower the *CER* value, the better the performance of the model. The perfect score for *CER* is 0, which indicates that no errors have been made.

2.2.6 Jaccard distance

Jaccard distance is not a metric dedicated to measuring speech recognition models. However, it can be effectively applied to this type of problem. It measures dissimilarity between sets. Its value can be calculated by dividing the difference between the sizes of the union and the intersection of two sets by the size of the union. The formula for *Jaccard distance* is:

$$JD(A,B) = (|A \cup B| - |A \cap B|)|A \cup B| = 1 - |A \cap B||A \cup B|$$

where:

A and B - two sets $|A|$ and $|B|$ - the size of A and B

$A \cap B$ - the intersection of A and B (the set of elements that are in both A and B)

$A \cup B$ - the union of A and B (the set of elements that are in either A or B or both)

Its value ranges from 0, which indicates complete similarity between the two sets to 1, indicating no similarity between sets.

3 Datasets

For this study, a new dataset has been developed with the intention of using it in the rapid testing of Polish-language medical speech recognition systems. This dataset does not constitute a new full-scale corpus of medical speech, it was created to standardize testing of selected tools. This choice is motivated by the lack of a publicly available medical corpus in Polish at this time. The purpose of this part of the research was not to create such a corpus. A full-scale corpus will be created in subsequent phases of the research, which are not part of this one. The dataset consists of 100 words, which were recorded individually. As a result, every actor produced 100 recordings. The words are medical terms in Polish, selected based on the International Classification of Diseases (ICD) dictionary version 9 PL. The dataset consists of 1200 recordings in total. 200 recordings were generated using synthesizer [13], 100 using female voice and 100 using male voice. The rest of the 1000 recordings were prepared by 10 actors (5 males and 5 females). The overview is presented in Table 1. A complete list of words and their translation and IPA (International Phonetic Alphabet) notation can be seen in the file provided as supplementary material to this paper, the shortened table image is presented in Fig.1.

#	PL	EN	IPA
1	adhezja	adhesion	adhɛzja
2	adrenalina	epinephrine	adrɛnalina
3	agregacja	aggregation	agrɛgat͡sja
4	aktywność	activity	aktivnɔɕt͡ɕ
5	aminokwasy	amino acids	aminɔkfasɨ
6	amoniak	ammonia	amɔɲak
7	amylaza	Amylase	amɨlaza
8	analityka	analytics	analitika
9	antygen	antigen	antigɛn
10	badanie	examination	badaɲɛ
11	białko	protein	bjawkɔ
12	chemia	chemistry	xɛmja
13	czynnik	factor	t͡ʂinɲik
...	...	...	...

Figure 1: List of sample words with translation and IPA notation

All of the participants were Polish natives, with Polish being their first language. All of the recordings were made using the home microphones available to them. This implies that the quality of the recordings varied between actors. The goal was not to test systems with clear recordings but to check their performance in a natural environment. No additional noise has been added. The Audacity free recording software [17] was chosen for recording. All of the recordings were stored in mp3 format. As a result of constituting only a single word, all recordings are short, with the average length being 1.37 seconds with a 0.37 standard deviation.

Table 1: Dataset summary

Subgroup	Number of samples
Number of samples (total)	1200
Number of samples (female voice)	600
Number of samples (male voice)	600
Number of samples (natural voice)	1000
Number of samples (synthetic voice)	200

4 Results

Using the predictions generated by the tools, as described in Section 2.1. Speech-to-text tools metrics were calculated. The tools are compared in terms of metrics detailed in section 2.2. Metrics. Scores for all tools were averaged based on gender and

natural/synthetic voice criterion - later named subsets of the data domain. The overall average is an average based on all 1200 speech samples. Female and Male Averages are the averages over 600 samples each, respectively recorded by women + generated by one female-voiced synthesizer (600 recordings) and men + generated by one male-voiced synthesizer (600 recordings). The natural average is calculated based on results for human-recorded samples (1000 recordings), and the synthetic average is computed using samples generated by a synthesizer (200 recordings). The following charts represent scores for each metric. The X-axis represents dataset groups, and the Y-axis represents the score that a specific model obtained. Tools and their variants are coded with the following colors:

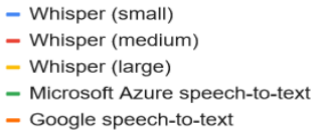

Figure 2 visualizes results for word error rate (*WER*). The best overall (Average), Average Female, Average Male, and Average Natural results were obtained by Google STT. In the case of synthetic recordings, the best results were gained by Microsoft Azure STT.

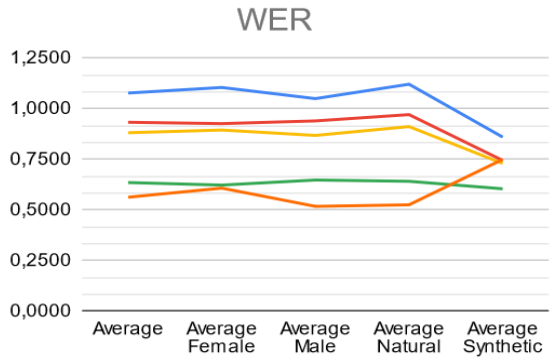

Figure 2: *WER* per STT tool per data domain subset

The visualization of results with regard to the Word Information Lost (*WIL*) rate metric is presented in Figure 3. Google STT obtains the best results for Average, Average Male, and Average Natural; Average Female and Average Synthetic best-performing tool is Microsoft Azure STT.

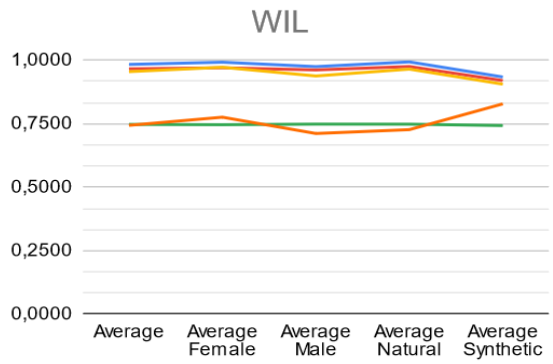

Figure 3: *WIL* per STT tool per data domain subset

The results based on *Levenshtein distance* and *Jaccard distance* are presented in Figures 4 and 5, respectively. For both, the best results were reached using Microsoft Azure STT.

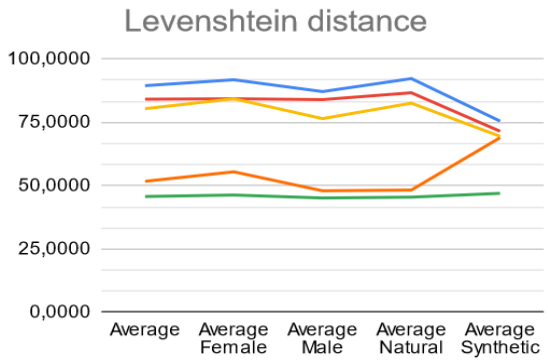

Figure 4: *Levenshtein distance* per STT tool per data domain subset

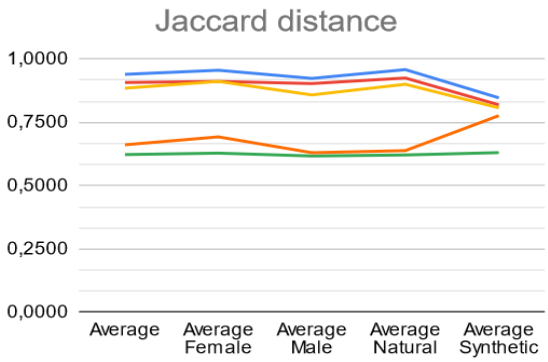

Figure 5: *Jaccard distance* per STT tool per data domain subset

Figure 6 presents results for Match Error Rate (*MER*). Again the finest results were obtained by Google STT and Microsoft Azure STT. The first one reports the best results for Average, Average Male, Average Natural, and the latter for Average Female, Average Synthetic.

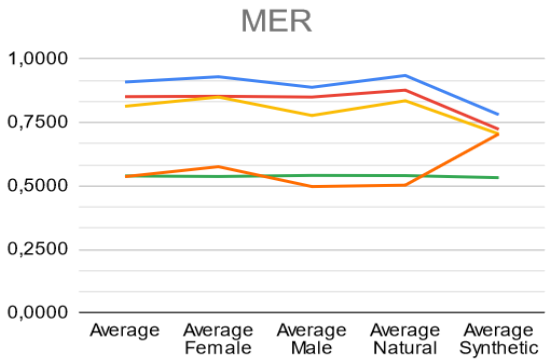

Figure 6: *MER* per STT tool per data domain subset

The last measure considered is Character Error Rate (*CER*). It is shown in Figure 8. The Microsoft Azure STT reached top results across all subsets of the data domain.

Metric	WER	WIL	Levenshtein distance	Jaccard distance	MER	CER
Score (Overall Average)	0,5633	0,7437	45,7500	0,9398	0,5369	0,1591
Score (Female Average)	0,6083	0,7462	46,3333	0,9559	0,5375	0,1713
Score (Male Average)	0,5183	0,7116	45,1667	0,9238	0,4978	0,1469
Score (Natural Average)	0,5260	0,7268	45,5000	0,9584	0,5033	0,1536
Score (Synthetic Average)	0,6050	0,7429	47,0000	0,8470	0,5328	0,1864

Microsoft Azure STT | Google STT

Figure 7: The best results per data domain subset

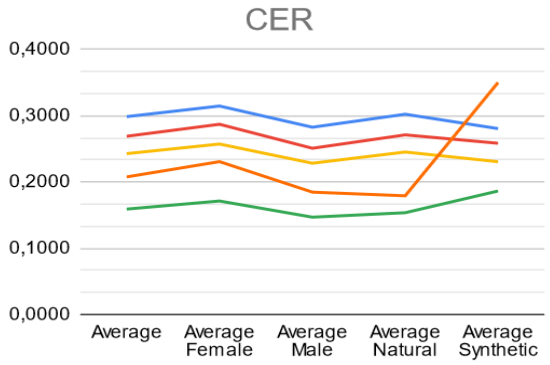

Figure 8: *CER* per STT tool per data domain subset

Figure 7 shows the best results for each data domain subset. The colors indicate the tool that was used to achieve specific results. The legend is presented below the table with the results. Color schema follows the one used in plots with overall results.

5 Conclusion

A new dataset for this study has been developed. The dataset consists of 100 sample words (medical domain) recorded by 10 actors and generated using 2 synthetic voices (in total 1200 recordings). This dataset was used to test well-known ASR tools for their ability to recognize Polish medical terms. The results have been gathered and compared between tools with respect to six performance metrics (*WER*, *WIL*, *Levenshtein distance*, *Jaccard distance*, *MER*, and *CER*). The Google STT and Microsoft Azure STT models performed better than Whisper. Google's model was best at recognizing natural voices and spikes to its worst results at synthetic voice transcription. On the other hand, Whisper models were best at recognizing synthetic voices but also revealed small drops in male voices. Microsoft Azure STT proved to be very consistent and has very small and rare spikes. It does not have big synthetic voice recognition spikes like Google STT and usually performs slightly better than Google STT. It can be concluded from the obtained results that all models better recognize male voices than female voices. After conducting these experiments, the authors are motivated to train a model that will allow us to develop a practical solution for the Polish medical language. To achieve this, a more complex dictionary with medical terminology will be created in cooperation with medical personnel. The dictionary will consist of whole expressions and not just individual words. One of the models tested and presented in this paper will be fine-tuned. The results presented in this paper show the magnitude of the problem which is the difficulty of recognizing medical speech in a not-so-common language by models that have been trained on data in that language.

Acknowledgements

This research was supported by the Polish National Centre for Research and Development (NCBR) within the project: "ADMED-VOICE - Adaptive intelligent speech processing system of medical personnel with the structuring of test results and support of therapeutic process". No. INFOSTRATEG4/0003/2022

References

[1] Distance package documentation. https://pypi.org/project/Distance/, (access 07.05.2023).
[2] Jiwer package documentation. https://pypi.org/project/jiwer/, (access 07.05.2023).
[3] Speech-to-text: Automatic speech recognition - google cloud, (access 07.05.2023). https://cloud.google.com/speech-to-text.
[4] Speech to text – audio to text translation | microsoft azure. https://azure.microsoft.com/en-us/products/cognitive-services/speech-to-text/, (access 07.05.2023).
[5] Graves A. Sequence transduction with recurrent neural networks. https://arxiv.org/abs/1211.3711.
[6] Kevin Chu, Leslie Collins, and Boyla Mainsah, 'Using automatic speech recognition and speech synthesis to improve the intelligibility of cochlear implant users in reverberant listening environments', in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6929–6933, (2020).
[7] Li Fu, Xiaoxiao Li, Libo Zi, Zhengchen Zhang, Youzheng Wu, Xiaodong He, and Bowen Zhou, 'Incremental learning for end-to-end automatic speech recognition', in *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 320–327, (2021).
[8] Ayoub Ghriss, Bo Yang, Viktor Rozgic, Elizabeth Shriberg, and Chao Wang, 'Sentiment-aware automatic speech recognition pre-training for enhanced speech emotion recognition', in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7347–7351, (2022).
[9] Google. Google stt documentation. https://cloud.google.com/speech-to-text/docs, (access 07.05.2023).
[10] Chiu Ch. Parmar N. Zhang Y. Yu J. Han W. Wang S. Zhang Z. Wu Y. Pang R. Gulati A., Qin J. Conformer: Convolution-augmented transformer for speech recognition. https://arxiv.org/abs/2005.08100.
[11] Casper J. Catanzaro B. Diamos G. Elsen E. Prenger R. Satheesh S. Sengupta S. Coates A. Ng A. Y. Hannun A., Case C. Deep speech: Scaling up end-to-end speech recognition. https://arxiv.org/abs/1412.5567.
[12] Shigeki Karita, Shinji Watanabe, Tomoharu Iwata, Marc Delcroix, Atsunori Ogawa, and Tomohiro Nakatani, 'Semi-supervised end-to-end speech recognition using text-to-speech and autoencoders', in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6166–6170, (2019).
[13] Bueno Ltd. Voice synthetiser speechgen.io. https://speechgen.io/pl/, (access 07.05.2023).
[14] Andrew Cameron Morris, Viktoria Maier, and Phil Green, 'From WER and RIL to MER and WIL: improved evaluation measures for connected speech recognition', in *Proc. Interspeech 2004*, pp. 2765–2768, (2004).

[15] OpenAI. Whisper, latest release. official github. https://github.com/openai/whisper/releases/tag/v20230314, (access 07.05.2023).
[16] Tao Xu Brockman G. McLeavey Ch. Sutskever I. Radford A., Kim W. J. Robust speech recognition via large-scale weak supervision. https://arxiv.org/abs/2212.04356.
[17] Audacity Team. Audacity documentation. https://www.audacityteam.org/, (access 07.05.2023).
[18] Punitha Vancha, Harshitha Nagarajan, Vishnu Sai Inakollu, Deepa Gupta, and Susmitha Vekkot, 'Word-level speech dataset creation for sourashtra and recognition system using kaldi', in *2022 IEEE 19th India Council International Conference (INDICON)*, pp. 1–6, (2022).
[19] Parmar N. Uszkoreit J. Jones L. Gomez A. N. Kaiser L. Polosukhin I. Vaswani A., Shazeer N. Attention is all you need. https://arxiv.org/abs/1706.03762.
[20] Wei Wang, Shuo Ren, Yao Qian, Shujie Liu, Yu Shi, Yanmin Qian, and Michael Zeng, 'Optimizing alignment of speech and language latent spaces for end-to-end speech recognition and understanding', in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7802–7806, (2022).