

A Crowdsourced Open-Source Kazakh Speech Corpus and Initial Speech Recognition Baseline

Yerbolat Khassanov, Saida Mussakhojayeva, Almas Mirzakhmetov,
Alen Adiyev, Mukhamet Nurpeiissov and Huseyin Atakan Varol

Institute of Smart Systems and Artificial Intelligence (ISSAI),

Nazarbayev University, Nur-Sultan, Kazakhstan

{yerbolat.khassanov, saida.mussakhojayeva, almas.mirzakhmetov,
alen.adiyev, mukhamet.nurpeiissov, ahvarol}@nu.edu.kz

Abstract

We present an open-source speech corpus for the Kazakh language. The Kazakh speech corpus (KSC) contains around 332 hours of transcribed audio comprising over 153,000 utterances spoken by participants from different regions and age groups, as well as both genders. It was carefully inspected by native Kazakh speakers to ensure high quality. The KSC is the largest publicly available database developed to advance various Kazakh speech and language processing applications. In this paper, we first describe the data collection and preprocessing procedures followed by a description of the database specifications. We also share our experience and challenges faced during the database construction, which might benefit other researchers planning to build a speech corpus for a low-resource language. To demonstrate the reliability of the database, we performed preliminary speech recognition experiments. The experimental results imply that the quality of audio and transcripts is promising (2.8% character error rate and 8.7% word error rate on the test set). To enable experiment reproducibility and ease the corpus usage, we also released an ESPnet recipe for our speech recognition models.

1 Introduction

We present an open-source Kazakh speech corpus (KSC) constructed to advance the development of speech and language processing applications for the Kazakh language. Kazakh is an agglutinative language with vowel harmony and belongs to the family of Turkic languages. During the Soviet period, the Kazakh language was overwhelmed by the Russian language, which caused a decline in Kazakh language usage (Dave, 2007). In the 1990s, it was declared an official language of Kazakhstan, and many initiatives were launched to increase the number of Kazakh speakers. Today, it is spoken by

over 10 million people in Kazakhstan and by over 3 million people in other countries¹. By introducing the KSC, we aim to accelerate the penetration of the Kazakh language into the Internet of things (IoT) technologies and to promote research in Kazakh speech processing applications.

Although several Kazakh speech corpora have been presented (Makhambetov et al., 2013; Shi et al., 2017; Mamyrbayev et al., 2019), there is no generally accepted common corpus. Most of them are either publicly unavailable or contain an insufficient amount of data to train reliable models. Especially, these databases are too small for building recent end-to-end models, which are extremely data hungry (Hannun et al., 2014). Consequently, different research groups usually conduct their experiments on internally collected data, which prevents the reproducibility and comparison of different approaches.

To address the aforementioned limitations, we created the KSC, containing around 332 hours of transcribed audio. It was crowdsourced through the Internet, where volunteers were asked to read sentences presented through a web browser. In total, we accepted over 153,000 utterances submitted from over 1,600 unique device IDs. The recordings were first checked manually, and when a sufficient amount of data was collected, were partially checked automatically using a speech recognition system. To the best of our knowledge, the KSC is the largest open-source speech corpus in Kazakh and is available for academic and commercial use upon request at this link² under the Creative Commons Attribution 4.0 International License³.

¹<https://www.ethnologue.com/language/kaz>

²<https://issai.nu.edu.kz/kz-speech-corpus/?version=1.1>

³<https://creativecommons.org/licenses/by/4.0/>

We expect that this database will be a valuable resource for research communities in both academia and industry. The primary application domains of the corpus are speech recognition, speech synthesis, and speaker recognition. To demonstrate the reliability of the database, we performed preliminary automatic speech recognition (ASR) experiments, where promising and sufficient for practical usage results were achieved. We also provide a practical guide on the development of ASR systems for the Kazakh language by sharing the reproducible recipe and pretrained models⁴. The utilization of the database for speech synthesis and speaker recognition tasks is left for future work.

The rest of the paper is organized as follows. Section 2 provides a review of related works. Section 3 presents the KSC database and explains the database construction procedures in detail. Section 4 offer a presentation of the speech recognition experiment setup and obtained results. In Section 5, the obtained results and challenges are discussed. Lastly, Section 6 concludes this paper and mentions potential future work.

2 Related Works

In the past few years, the interest in ASR has surged by its new applications in smart devices, such as voice command, voice search, message dictation, and virtual assistants (Yu and Deng, 2014). In response to this technological shift, many speech corpora have been introduced for various languages. For example, Du et al. (2018) released 1,000 hours of open-source Mandarin read speech data to bridge the gap between academia and industry. The utterances were recorded using iOS-based mobile phones and cover the following domains: voice command, smart home, autonomous driving, and so on. Similarly, Koh et al. (2019) developed a 2,000-hour read speech corpus to help speech technology developers and researchers build speech-related applications in Singapore.

Several works have presented speech databases for the Kazakh language. For example, Makhambetov et al. (2013) developed a Kazakh language corpus containing around 40 hours of transcribed read speech data recorded in a sound-proof studio. Similarly, Mamyrbayev et al. (2019) collected 76 hours of data using a professional recording booth which were further extended to 123 hours

⁴https://github.com/IS2AI/ISSAI_SAIDA_Kazakh_ASR

in Mamyrbayev et al. (2020). Khomitsevich et al. (2015) utilized 147 hours of bilingual Kazakh-Russian speech data to build code-switching ASR systems. Shi et al. (2017) released 78 hours of transcribed Kazakh speech data recorded by 96 students from China. The IARPA Babel project has released a Kazakh language pack⁵ consisting of around 50 hours of conversational and 14 hours of scripted telephone speech. Unfortunately, the aforementioned databases are either publicly unavailable or of an insufficient size to build robust Kazakh ASR systems. Additionally, some of them are nonrepresentative—that is, they cover speakers from a narrow set of categories, such as the same region or age group. Furthermore, since most of these databases have been collected in optimal lab settings, they might be ineffective for real-world applications.

The emergence of crowdsourcing platforms and the growth in Internet connectivity has motivated researchers to employ crowdsourcing for annotated corpora construction. Different from the expert-based approaches, crowdsourcing tends to be cheaper and faster, though, additional measures should be taken to ensure the data quality (Snow et al., 2008; Novotney and Callison-Burch, 2010; Eskenazi et al., 2013). Furthermore, crowdsourcing allows to gather a variety of dialects and accents from remote geographical locations and enables the participation of people with disabilities and of an advanced age, which otherwise would be impossible or too costly (Takamichi and Saruwatari, 2018). Inspired by this, we followed the best crowdsourcing practices to construct a large-scale and open-source speech corpus for the Kazakh language, as described in the following sections.

3 The KSC Construction

The KSC project was conducted with the approval of the Institutional Research Ethics Committee of Nazarbayev University. Each reader participated voluntarily and was informed of the data collection and use protocols through an online consent form.

3.1 Text Collection and Cleaning

We first extracted Kazakh textual data from various sources such as electronic books, laws, and websites, including Wikipedia, news portals, and blogs. For each website, we designed a specialized

⁵<https://catalog ldc.upenn.edu/LDC2018S13>

web crawler to improve the quality of the extracted text. The extracted texts were manually filtered to eliminate inappropriate content involving sensitive political issues, user privacy, violence, and so on. Additionally, we filtered out texts entirely consisting of Russian words. Texts consisting of mixed Kazakh-Russian utterances were kept, because there are many borrowed Russian words in Kazakh, and it is common practice among Kazakh speakers to code-switch between Kazakh and Russian (Khomitsevich et al., 2015). Next, we split the texts into sentences and removed sentences consisting of more than 25 words. Lastly, duplicate sentences were removed. The total number of extracted sentences was around 2.3 million.

3.2 Text Narration and Checking

To narrate the extracted sentences, we developed a web-based speech recording platform capable of running on personal computers and smartphones. The platform randomly samples a sentence from the pool of extracted texts, and presents it to a reader (see Figure 1). It also displays the recording status and statistics, such as elapsed time and the total number of read sentences. Additionally, it has the “pause” and “next” buttons to control the recording process. The readers were allowed to quit at any time. We recruited readers by advertising the project in social media, news, and open messaging communities on WhatsApp and Telegram. Readers who were at least 18 years old were included so that they could legally agree to participate in data collection. The audios were recorded in 48 kHz and 16 bits, but downsampled to 16 kHz and 16 bits for online publication. Following our experimental protocol, we did not store readers’ personal

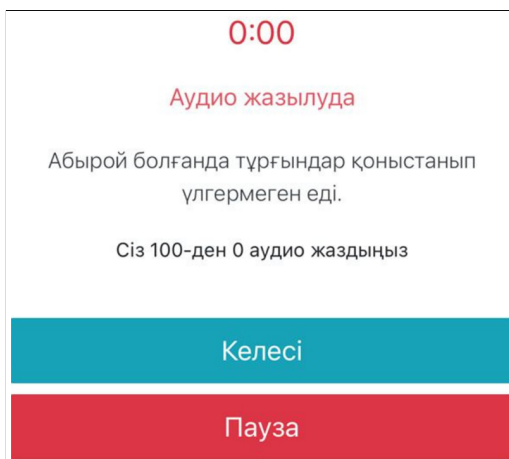


Figure 1: The speech recording web interface.

Category	Train	Valid	Test	Total
Duration (hours)	318.4	7.1	7.1	332.6
# Utterances	147,236	3,283	3,334	153,853
# Words	1.61M	35.2k	35.8k	1.68M
# Unique Words	157,191	13,525	13,959	160,041
# Device IDs	1,554	29	29	1,612
# Speakers	-	29	29	-

Table 1: The KSC database specifications.

information except for the geolocation coordinates, IP address, and device type.

Several native Kazakh transcribers were hired to check the quality of recordings. The transcribers logged on to a special transcription checking platform and were provided with an audio segment and the corresponding sentence that a reader had read. The task was to check if the reader had read the sentence according to the prompt, and to transcribe any deviations or other acoustic events based on a set of transcription instructions. As an additional quality measure, we hired a linguist who was assigned to supervise the transcribers and to randomly check tasks they completed. To harmonize the transcriptions, the linguist also held the “go through errors” sessions with the transcribers.

The transcribers were instructed to reject utterances containing obvious mispronunciations or severe noises, to convert numbers into words⁶, and to trim long silences at the beginning and end of the audio segments. Additionally, they were instructed to enclose partial repetitions and hesitations in parentheses, for example, ‘(he) hello’ and ‘(ah)’, and to indicate other non-verbal sounds produced by readers, such as sneezing and coughing, using a special ‘[noise]’ token. Background noises were not labeled.

When the size of accepted utterances reached 100 hours, we built an ASR system to automatically check the recordings. The system accepted only recordings perfectly matching corresponding text prompts—that is, 0% character error rate (CER), whereas the remaining utterances were left to human transcribers.

3.3 Database Specifications

The KSC database specifications are provided in Table 1. We split the data into three sets of non-overlapping speakers: training, validation, and test. While the training set recordings were collected

⁶Note that after converting numbers into words some sentence lengths exceeded 25 words.

Category		Valid	Test
Gender (%)	Female	51.7	51.7
	Male	48.3	48.3
Age (%)	18-27	37.9	34.5
	28-37	34.5	31.0
	38-47	10.4	13.8
	48 and above	17.2	20.7
Region (%)	East	13.8	13.8
	West	20.7	17.2
	North	13.8	20.7
	South	37.9	41.4
	Center	13.8	6.9
Device (%)	Phone	62.1	79.3
	Computer	37.9	20.7
Headphone (%)	Yes	20.7	17.2
	No	79.3	82.8

Table 2: The validation and test sets speaker details.

from anonymous speakers, the validation and test sets were collected from identifiable speakers to ensure that they did not overlap with the training set, represented different age groups and regions, and were gender balanced (see Table 2). In total, around 153,000 utterances were accepted, yielding 332 hours of transcribed speech data. Note that the device IDs could not be used to represent the number of speakers in the training set as several speakers might have used the same device or the same speaker might have used different devices. Therefore, the total number of speakers in the training set is not shown in Table 1. The whole database creation process took around four months, and the database size is around 38GB.

The Kazakh writing system differs depending on the regions where the language is spoken. For example, the Cyrillic alphabet is used in Kazakhstan and Mongolia, while an Arabic-derived alphabet is used in China. In the KSC, we presented all texts using the Cyrillic alphabet consisting of 42 letters. The distribution of these letters in the KSC is given in Figure 2.

One of the important features of the proposed KSC database is that it was collected in various environment conditions (e.g. home, office, café, transport, and street), with diverse background noises through mobile devices (e.g. phones and tablets) and personal computers, with and without headphone sets, which is similar to realistic use-case scenarios. Consequently, our database enables the development and evaluation of ASR systems designed to operate in real-world voice-enabled applications, such as voice commands, voice search, message dictation, and so on.

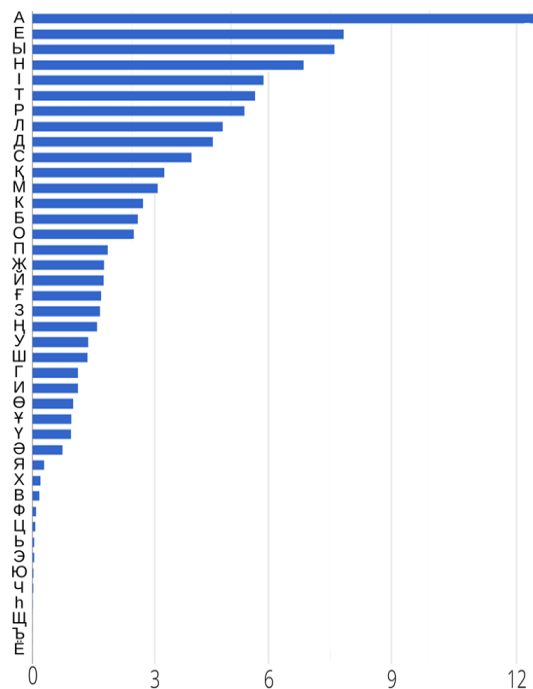


Figure 2: The distribution of letters in the KSC (%).

The KSC database consists of audio recordings, transcripts, and metadata stored in separate folders. The audio and corresponding transcription file names are the same, except that the audio recordings are stored as WAV files, whereas the transcriptions are stored as TXT files using the UTF-8 encoding. The metadata contain the data splitting information (training, validation, and test) and the speaker’s details (gender, age, region, device, and headphones) of the validation and test sets.

4 Speech Recognition Experiment

To demonstrate the utility and reliability of the KSC database, we conducted speech recognition experiments using both the traditional deep neural network-hidden Markov model (DNN-HMM) and recently proposed end-to-end (E2E) architectures. We did not compare or perform thorough architecture searches for either DNN-HMM and E2E, since this falls outside the scope of this paper.

4.1 Experimental Setup

All ASR models were trained with a single V100 GPU running on the NVIDIA DGX-2 server using the training set. All hyper-parameters were tuned using the validation set. The best-performing model was evaluated using the test set. All results are reported without lattice or n-best hypotheses rescoring, and no external data have been used.

4.1.1 The DNN-HMM ASR

The DNN-HMM ASR system was built using the Kaldi framework (Povey et al., 2011). We followed the Wall Street Journal (WSJ) recipe with the “nnet3+chain” setup and other latest Kaldi developments. The acoustic model was constructed using the factorized time-delay neural networks (TDNN-F) (Povey et al., 2018) trained with the lattice-free maximum mutual information (LF-MMI) (Povey et al., 2016) training criterion. The inputs were Mel-frequency cepstral coefficients (MFCC) features with cepstral mean and variance normalization extracted every 10 ms over a 25 ms window. In addition, we applied data augmentation using the speed perturbation (SpeedPerturb) (Ko et al., 2015) at rates of 0.9, 1.0, and 1.1 and the spectral augmentation (SpecAugment) (Park et al., 2019) techniques.

We employed a graphemic lexicon because of the strong correspondence between word spelling and pronunciation in Kazakh (e.g. “*hello* → *h e l l o*”). The graphemic lexicon was constructed by extracting all words in the training set, which resulted in 157,191 unique words. During the decoding stage, we employed a 3-gram language model⁷ (LM) with the Kneser-Ney smoothing built using the SRILM toolkit (Stolcke, 2002). The 3-gram LM was trained using the transcripts of the training set and the vocabulary covering all the words in the graphemic lexicon.

4.1.2 The E2E ASR

The E2E ASR systems were built using the ESPnet framework (Watanabe et al., 2018). We followed the WSJ recipe to train two different encoder-decoder architectures based on the recurrent neural networks (RNN) (Bahdanau et al., 2015) and the Transformer (Vaswani et al., 2017). Both architectures were jointly trained with the connectionist temporal classification (CTC) (Graves et al., 2006) objective function under the multi-task learning framework (Kim et al., 2017). The input speech was represented as an 80-dimensional filterbank features with pitch computed every 10 ms over a 25 ms window. For both E2E architectures, the acoustic features were first processed by few initial blocks of VGG network (Simonyan and Zisserman, 2015). Since Kazakh is a morphologically rich language, it is susceptible to severe data sparseness.

⁷We tested different N-gram orders and found the 3-gram to perform best on our setup.

To overcome this issue, we employed character-level output units in both architectures. In total, we used 45 distinct output units consisting of 42 letters from the Kazakh alphabet and 3 special tokens—that is, $\langle unk \rangle$, $\langle space \rangle$, and $\langle blank \rangle$ used in CTC.

The E2E ASR systems do not require a lexicon when modeling with grapheme-based output units (Sainath et al., 2018). The character-level LM was built using the transcripts of the training set as a two-layer RNN with 650 long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) units each. We utilized the LSTM LM during the decoding stage using the shallow fusion (Gülçehre et al., 2015) for both E2E architectures. Besides, we augment the training data using the SpeedPerturb and the SpecAugment techniques. For decoding, we set the beam size to 30 and the LSTM LM interpolation weight to 1.

E2E-RNN. The encoder module of the RNN-based E2E ASR system consists of three bi-directional LSTM layers with 1,024 units per direction per layer. The decoder module is a single uni-directional LSTM with 1,024 units. We train the model for 20 epochs using the Adadelta optimizer with the initial learning rate set to 1 and the batch size set to 30. The interpolation weight for the CTC objective was set to 0.5.

E2E-Transformer. The Transformer-based E2E ASR system consists of 12 encoder and 6 decoder blocks. We set the number of heads in the self-attention layer to 4 with 256-dimension hidden states and the feed-forward network dimensions to 2,048. We set the dropout rate and label smoothing to 0.1. The model was trained for 160 epochs using the Noam optimizer (Vaswani et al., 2017) with the initial learning rate of 10 and the warmup-steps of 25,000. The batch size was set to 96. We report results on an average model constructed using the last 10 checkpoints. The interpolation weight for the CTC objective was set to 0.3.

4.2 Experimental Results

The experimental results are presented in Table 3 in terms of both the character error rate (CER) and word error rate (WER). All the ASR models achieved competitive results on both the validation and test sets. We found that the validation set is more challenging than the test set. When compared without SpecAugment, the performance of the DNN-HMM model (ID 1) is slightly better than the E2E-RNN (ID 5), but inferior to the

ID	Model	LM	SpeedPerturb	SpecAugment	Valid		Test	
					CER	WER	CER	WER
1	DNN-HMM	Yes	Yes	No	5.2	14.2	4.6	13.7
2		Yes	Yes	Yes	5.3	14.9	4.7	13.8
3	E2E-LSTM	No	No	No	9.9	32.0	8.7	28.8
4		Yes	No	No	7.9	20.1	7.2	18.5
5		Yes	Yes	No	5.7	15.9	5.0	14.4
6		Yes	Yes	Yes	4.6	13.1	4.0	11.7
7	E2E-Transformer	No	No	No	6.1	22.2	4.9	18.8
8		Yes	No	No	4.5	13.9	3.7	11.9
9		Yes	Yes	No	3.9	12.3	3.2	10.5
10		Yes	Yes	Yes	3.2	10.0	2.8	8.7

Table 3: The CER (%) and WER (%) performances of different ASR models built using KSC.

E2E-Transformer (ID 9). We could not achieve any improvements on the DNN-HMM model (ID 2) using the SpecAugment despite trying different hyper-parameter tuning recommendations (Zhou et al., 2020). Overall, the best CER and WER results are achieved by the E2E-Transformer (ID 10) followed by the E2E-RNN (ID 6) and then the DNN-HMM (ID 1).

We observed that the LM fusion significantly improves the performances of both E2E models. For example, 35% and 36% relative WER improvements are achieved on the test set for the RNN (from ID 3 to ID 4) and Transformer (from ID 7 to ID 8) models, respectively. Furthermore, both data augmentation techniques based on SpeedPerturb and SpecAugment are highly effective for the Kazakh E2E ASR where additional improvements are achieved. For example, when models without and with data augmentations are compared, 36% and 26% relative WER improvements are achieved on the test set for the RNN (from ID 4 to ID 6) and Transformer (from ID 8 to ID 10) models, respectively.

These experimental results successfully demonstrate the utility of the KSC database for the speech recognition task. We leave the exploration of the optimal hyper-parameter settings and detailed comparison of different ASR architectures for future work.

5 Discussion

1) *Data sparsity*. Kazakh language speech recognition is considered challenging due to the agglutinative nature of the language, where word structures

are formed by adding derivational and inflectional affixes to stems in a specific order. As a result, the vocabulary size might considerably grow resulting in a data sparsity problem, especially for models operating in a word level, such as our DNN-HMM architecture. The potential solution is to break down words into finer-level linguistic units, such as characters or subword units (Sennrich et al., 2016). We investigated the impact of other output unit sizes on the performance of the Kazakh E2E-Transformer ASR and did not observe any considerable improvements over the character-level outputs (see Figure 3). The output units were generated using the byte pair encoding (BPE) algorithm implemented in the SentencePiece (Kudo and Richardson, 2018) tokenizer.

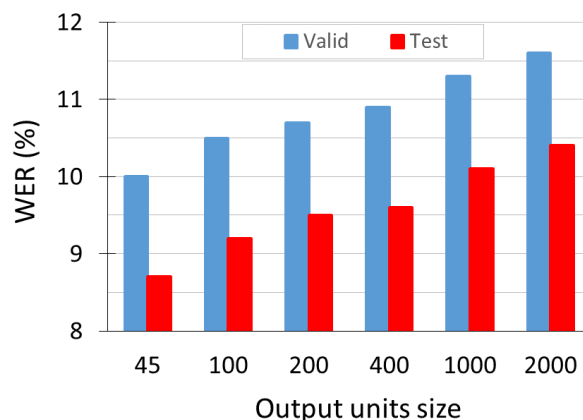


Figure 3: The impact of different output unit sizes on the E2E-Transformer ASR performance.

2) *Code-switching*. Another challenge is the Kazakh-Russian code-switching practice which is common in daily communication as the majority of

ID	Confusion pairs	Insertion	Deletion	Substitution
1	жоғарғы ⇒ жоғары (jogargy ⇒ jogary) “upper” or “higher” ⇒ “top” or “high”	мен (men) “I” or “with”	де (de) “too” (after a thin-vowelled word)	да (da) “too” (after a thick-voweled word)
2	өзеннің ⇒ өзенінің (uzennin ⇒ uzeninin) “of a river” ⇒ “of the river”	бұл (bul) “this” or “these”	ал (al) an imperative of “take” or “whereas”	де (de) “too” (after a thin-vowelled word)
3	ас ⇒ ақ (as ⇒ aq) “meal” ⇒ “white”	бір (bir) “one”	да (da) “too” (after a thick-vowelled word)	мен (men) “I” or “with”
4	бағы ⇒ баға (bagy ⇒ бага) “the garden” ⇒ “price”	да (da) “too” (after a thick-vowelled word)	әр (ar) “every”	жылы (jyly) “the year” or “warm”
5	болу ⇒ болуы (bolu ⇒ boluy) “being” ⇒ “the being”	ақ (aq) “white”	бір (bir) “one”	бір (bir) “one”

Table 4: The top five confusion pairs (reference word ⇒ recognized word) and insertion, deletion, and substitution errors in the recognized output of the E2E-Transformer ASR.

Kazakhs are bilingual. Mostly, inter-sentential and intra-sentential types of code-switching are practiced, however, intra-word code-switching is also possible. For example, one can say “*Men magazine bardym*” (“*I went to a store*”), where the Russian word “*magazin*” is appended by the Kazakh inflection “-ge” representing the preposition “to”. Furthermore, while the spelling of Kazakh words closely matches their pronunciation, this is not the case for Russian words, for example, the letter “o” is sometimes pronounced as /a/, which might confuse an ASR system. We observed that our ASR system is ineffective in code-switched utterances. Therefore, future work should focus on alleviating these errors.

3) *Data efficiency*. To analyze the data efficiency—that is, increase in performance due to the addition of new training data—we trained our best E2E-Transformer ASR system using different amounts of data. In particular, we first randomly sampled 40 hours of data and kept increasing their size until the entire training set was covered. The experimental results indicate that the WER performance has not converged yet and further data collection might be effective (see Figure 4). Therefore, we plan to continue the data collection process.

4) *ASR output analysis*. We inspected the recognized outputs of the best E2E-Transformer model to identify the most challenging characters and words. Table 4 lists the top five confusion pairs and insertion, deletion, and substitution errors. Most of them are commonly used words, such as conjunctions and numbers. In addition, we also inspected the most confused character pairs (see Figure 5). We observed that the Kazakh ASR system confuses characters with a similar pronunciation, such as “н” (/n/) and “ң” (/ŋ/), “і” (/ɪ/) and “ы” (/ə/), and so on.

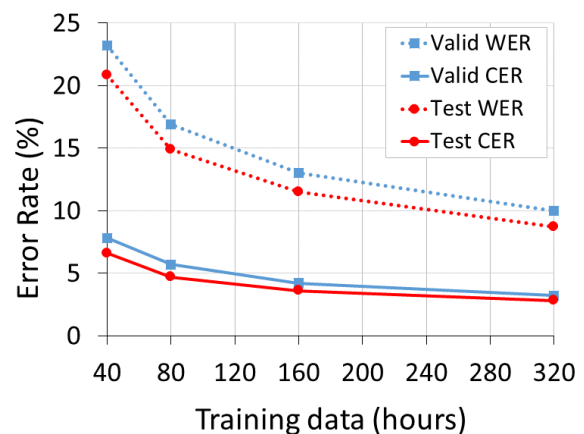


Figure 4: The data efficiency experiment.

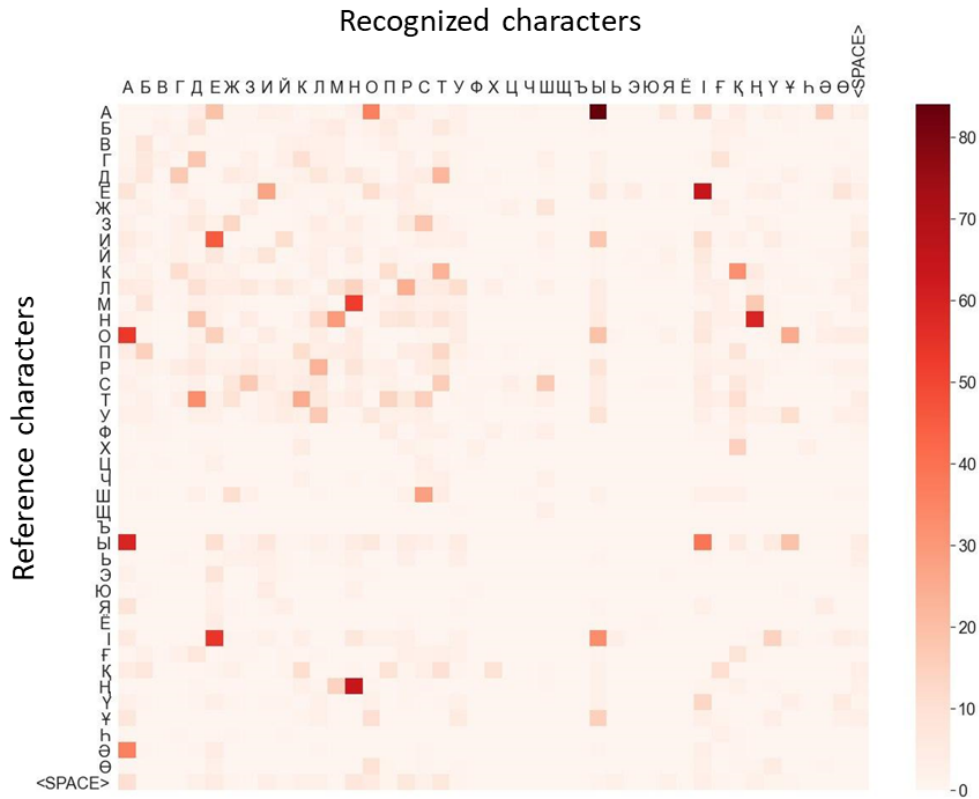


Figure 5: The character confusion matrix for the E2E-Transformer ASR.

5) *Performance comparison.* We cannot directly compare our results to previous works; however, our WER results are appealing. For example, [Mamyrbayev et al. \(2019\)](#) used 76 hours of speech data to build an DNN-HMM ASR system which achieved 32.7% WER on clean read speech. Similarly, the DNN-HMM ASR system built using 78 hours of data in ([Shi et al., 2017](#)) achieved 25.1% WER on read speech. On the other hand, [Mamyrbayev et al. \(2020\)](#) achieved 17.8% WER on clean read speech using the E2E ASR system trained on 126 hours of data. In comparison, our best model achieved 8.7% WER on the test set.

6) *Benefit for other Turkic Languages.* We also envision that the KSC can be utilized in cross-lingual transfer learning techniques ([Das and Hasegawa-Johnson, 2015](#)) to improve ASR systems for other Turkic languages, such as Kyrgyz and Tatar.

6 Conclusion

In this work, we presented the KSC database containing around 332 hours of transcribed speech data. It was developed to advance Kazakh speech processing applications, such as speech recognition,

speech synthesis, and speaker recognition. We described the database construction procedures and discussed challenges that should be addressed in future work. The described methodologies might benefit other researchers planning to build a speech corpus for a low-resource language. The database is freely available for any purposes including research and commercial use. We also conducted preliminary speech recognition experiments using both traditional hybrid DNN-HMM and recently proposed E2E ASR architectures. To ease the database usage and ensure the reproducibility of experiments, we split it into three non-overlapping sets (training, validation, and test) and released our ESPnet recipe. The detailed exploration of better ASR settings, as well as the adaptation of the database to other applications, is left for future work.

Acknowledgments

We would like to thank Aigerim Boranbayeva for helping to train and monitor the transcribers. We also thank Kuralay Baimenova, Yerbol Absalyamov, Gibrat Kurmanov, and Rustem Yeshpanov for helping to attract readers and to resolve other administrative tasks.

References

- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. [Neural machine translation by jointly learning to align and translate](#). In *Proc. of the 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, May 7-9, 2015*.
- Amit Das and Mark Hasegawa-Johnson. 2015. [Cross-lingual transfer learning during supervised training in low resource scenarios](#). In *Proc. of the 16th Annual Conference of the International Speech Communication Association (INTERSPEECH), Dresden, Germany, September 6-10, 2015*, pages 3531–3535.
- Bhavna Dave. 2007. *Kazakhstan-ethnicity, language and power*. Routledge.
- Jiayu Du, Xingyu Na, Xuechen Liu, and Hui Bu. 2018. [AISHELL-2: transforming mandarin ASR research into industrial scale](#). *CoRR*, abs/1808.10583.
- Maxine Eskenazi, Gina-Anne Levow, Helen Meng, Gabriel Parent, and David Suendermann. 2013. *Crowdsourcing for speech processing: Applications to data collection, transcription and assessment*. John Wiley & Sons.
- Alex Graves, Santiago Fernández, Faustino J. Gomez, and Jürgen Schmidhuber. 2006. [Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks](#). In *Proc. of the 23rd International Conference on Machine Learning, (ICML), Pittsburgh, Pennsylvania, USA, June 25-29, 2006*, pages 369–376.
- Çaglar Gülçehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loïc Barrault, Hui-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. [On using monolingual corpora in neural machine translation](#). *CoRR*, abs/1503.03535.
- Awni Y. Hannun, C. Case, J. Casper, Bryan Catanzaro, Greg Diamos, E. Elsen, Ryan Prenger, S. Satheesh, S. Sengupta, A. Coates, and A. Ng. 2014. [Deep speech: Scaling up end-to-end speech recognition](#). *ArXiv*, abs/1412.5567.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. [Long short-term memory](#). *Neural Comput.*, 9(8):1735–1780.
- Olga Khomitshevich, Valentin Mendelev, Natalia A. Tomashenko, Sergey Rybin, Ivan Medennikov, and Saule Kudubayeva. 2015. [A bilingual kazakh-russian system for automatic speech recognition and synthesis](#). In *Proc. of the 17th International Conference on Speech and Computer (SPECOM), Athens, Greece, September 20-24, 2015*, volume 9319, pages 25–33.
- Suyoun Kim, Takaaki Hori, and Shinji Watanabe. 2017. [Joint ctc-attention based end-to-end speech recognition using multi-task learning](#). In *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, March 5-9, 2017*, pages 4835–4839.
- Tom Ko, Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur. 2015. [Audio augmentation for speech recognition](#). In *Proc. of the 16th Annual Conference of the International Speech Communication Association (INTERSPEECH), Dresden, Germany, September 6-10, 2015*, pages 3586–3589.
- Jia Xin Koh, Aqilah Mislán, Kevin Khoo, Brian Ang, Wilson Ang, Charmaine Ng, and Ying-Ying Tan. 2019. [Building the singapore english national speech corpus](#). In *Proc. of the 20th Annual Conference of the International Speech Communication Association (INTERSPEECH), Graz, Austria, 15-19 September 2019*, pages 321–325.
- Taku Kudo and John Richardson. 2018. [Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018*, pages 66–71. Association for Computational Linguistics.
- Olzhas Makhambetov, Aibek Makazhanov, Zhandos Yessenbayev, Bakhyt Matkarimov, Islam Sabyrgaliyev, and Anuar Sharafudinov. 2013. [Assembling the kazakh language corpus](#). In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 18-21 October 2013, Seattle, Washington, USA*, pages 1022–1031.
- Orken Mamyrbayev, Keylan Alimhan, Bagashar Zhuzmazhanov, Tolganay Turdalykyzy, and Farida Gusmanova. 2020. [End-to-end speech recognition in agglutinative languages](#). In *Proc. of the 12th Asian Conference on Intelligent Information and Database Systems (ACIIDS), Phuket, Thailand, March 23-26, 2020*, volume 12034, pages 391–401.
- Orken J. Mamyrbayev, Mussa Turdalyuly, Nurbapa Mekebayev, Keylan Alimhan, Aizat Kydyrbekova, and Tolganay Turdalykyzy. 2019. [Automatic recognition of kazakh speech using deep neural networks](#). In *Proc. of the 11th Asian Conference on Intelligent Information and Database Systems (ACIIDS), Yogyakarta, Indonesia, April 8-11, 2019*, pages 465–474.
- Scott Novotney and Chris Callison-Burch. 2010. [Cheap, fast and good enough: Automatic speech recognition with non-expert transcription](#). In *Proc. of Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, June 2-4, 2010, Los Angeles, California, USA*, pages 207–215.
- Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. [SpecAugment: A simple data augmentation method for automatic speech recognition](#). In *Proc. of the 20th Annual Conference of the International Speech Communication Association (INTERSPEECH), Graz, Austria, 15-19 September 2019*, pages 2613–2617.

- Daniel Povey, Gaofeng Cheng, Yiming Wang, Ke Li, Hainan Xu, Mahsa Yarmohammadi, and Sanjeev Khudanpur. 2018. [Semi-orthogonal low-rank matrix factorization for deep neural networks](#). In *Proc. of the 19th Annual Conference of the International Speech Communication Association (INTERSPEECH), Hyderabad, India, 2-6 September 2018*, pages 3743–3747.
- Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al. 2011. The Kaldi speech recognition toolkit. In *Proc. of the IEEE Workshop on Automatic Speech Recognition and Understanding*.
- Daniel Povey, Vijayaditya Peddinti, Daniel Galvez, Pegah Ghahremani, Vimal Manohar, Xingyu Na, Yiming Wang, and Sanjeev Khudanpur. 2016. [Purely sequence-trained neural networks for ASR based on lattice-free MMI](#). In *Proc. of the 17th Annual Conference of the International Speech Communication Association (INTERSPEECH), San Francisco, CA, USA, September 8-12, 2016*, pages 2751–2755.
- Tara N. Sainath, Rohit Prabhavalkar, Shankar Kumar, Seungji Lee, Anjuli Kannan, David Rybach, Vlad Schogol, Patrick Nguyen, Bo Li, Yonghui Wu, Zhifeng Chen, and Chung-Cheng Chiu. 2018. [No need for a lexicon? Evaluating the value of the pronunciation lexica in end-to-end models](#). In *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, April 15-20, 2018*, pages 5859–5863.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. [Neural machine translation of rare words with subword units](#). In *Proc. of the 54th Annual Meeting of the Association for Computational Linguistics (ACL), August 7-12, 2016, Berlin, Germany*.
- Ying Shi, Askar Hamdullah, Zhiyuan Tang, Dong Wang, and Thomas Fang Zheng. 2017. [A free Kazakh speech database and a speech recognition baseline](#). In *Proc. of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), Kuala Lumpur, Malaysia, December 12-15, 2017*, pages 745–748.
- Karen Simonyan and Andrew Zisserman. 2015. [Very deep convolutional networks for large-scale image recognition](#). In *Proc. of the 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, May 7-9, 2015*.
- Rion Snow, Brendan O’Connor, Daniel Jurafsky, and Andrew Y. Ng. 2008. [Cheap and fast - But is it good? Evaluating non-expert annotations for natural language tasks](#). In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 25-27 October 2008, Honolulu, Hawaii, USA*, pages 254–263.
- Andreas Stolcke. 2002. [SRILM - An extensible language modeling toolkit](#). In *Proc. of the International Conference on Spoken Language Processing (ICSLP), Denver, Colorado, USA, September 16-20, 2002*.
- Shinnosuke Takamichi and Hiroshi Saruwatari. 2018. [CPJD corpus: Crowdsourced parallel speech corpus of Japanese dialects](#). In *Proc. of the 11th International Conference on Language Resources and Evaluation (LREC), Miyazaki, Japan, May 7-12, 2018*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In *Proc. of the Annual Conference on Advances in Neural Information Processing Systems (NIPS), 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008.
- Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplín, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai. 2018. [ESPnet: End-to-end speech processing toolkit](#). In *Proc. of the 19th Annual Conference of the International Speech Communication Association (INTERSPEECH), Hyderabad, India, 2-6 September 2018*, pages 2207–2211.
- Dong Yu and Li Deng. 2014. *Automatic Speech Recognition: A Deep Learning Approach*. Springer Publishing Company.
- Wei Zhou, Wilfried Michel, Kazuki Irie, Markus Kitzka, Ralf Schlüter, and Hermann Ney. 2020. [The rwth asr system for ted-lium release 2: Improving hybrid hmm with specaugment](#). In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020*, pages 7839–7843. IEEE.