INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

MOCKS 1.0: Multilingual Open Custom Keyword Spotting Testset

Mikołaj Pudo[1,2], *Mateusz Wosik*[1], *Adam Cieślak*[1], *Justyna Krzywdziak*[1], *Bożena Łukasiak*[1], *Artur Janicki*[2]

[1]Samsung R&D Institute Poland
[2]Warsaw University of Technology, Poland

m.pudo@samsung.com, m.wosik@samsung.com, a.cieslak@samsung.com,
j.krzywdziak@partner.samsung.com, b.lukasiak@samsung.com, artur.janicki@pw.edu.pl

Abstract

The main purpose of this work is to create a comprehensive audio testset that can be used to evaluate custom keyword spotting (KWS) models and to benchmark different KWS solutions. We also propose a set of requirements that should be followed while creating testsets to evaluate custom KWS models. We consider multiple versions of the problem: text and audio-based keyword spotting, as well as offline and online (streaming) modes. Our testset named MOCKS is based on LibriSpeech and Mozilla Common Voice datasets. We used automatically generated alignments to extract parts of the recordings, which were split into keywords and test samples. The resulting testset contains almost 50,000 keywords. It contains audio data in English, French, German, Italian, and Spanish, but can be easily extended to other languages. MOCKS has been made publicly available to the research community. Initial KWS experiments run on MOCKS suggest that it can serve as a challenging testset for future research.

Index Terms: keyword spotting, custom keyword spotting, query-by-text, query-by-example, corpus design, multilingual audio corpus

1. Introduction

Voice interaction with electronic devices has become a standard for various tasks replacing keyboards. The most important examples where such type of interaction is useful are intelligent assistants. At the beginning of this revolution, each conversation would start with a dedicated button (push-to-talk solution), but soon the hands-free option was introduced. Currently, all leading intelligent assistants use keyword detection for conversation initialization. This task involves deciding whether an audio recording contains a speech signal similar enough to a keyword (text or audio). The problem of keyword spotting comes in two flavors:

1. custom keyword is provided by text (in literature: *query-by-text keyword spotting* or QbyT KWS for short),
2. custom keyword is provided by audio (in literature: *query-by-example keyword spotting* or QbyE KWS for short).

Throughout this paper, KWS will mean both QbyT KWS and QbyE KWS tasks. Various attempts to solve KWS have been proposed. Unfortunately, many of the solutions were evaluated on the proprietary testsets, making a comparison between models impossible. There is only a small number of public testsets that can be used to evaluate KWS models, such as [1, 2, 3, 4]. However, those testsets do not allow for an in-depth evaluation of the models since they contain a limited number of keywords and they lack challenging negative examples of keywords.

In this paper, we describe our attempt to fill in this gap by proposing a public testset built upon data from LibriSpeech [5] and Mozilla Common Voice (MCV) [6]. We named it MOCKS: *Multilingual Open Custom Keyword Spotting Testset*, and we made it freely available at `https://huggingface.co/datasets/voiceintelligenceresearch/MOCKS`. Therefore, we believe that such a testset can become an effective tool when evaluating or benchmarking KWS algorithms.

The rest of this paper is organized as follows. In Section 2, we describe a selection of the solutions to the custom keyword spotting problem and the available testsets. In Section 3, we describe the requirements which should be followed during corpus preparation. In Section 4, we provide a detailed description of our corpus. We also present baseline evaluation results in Section 5. Finally, we summarize our paper in Section 6.

2. Related work

Both versions of KWS tasks, text-based and audio-based, have been approached before. A review of the solutions to QbyT KWS, accompanied by a list of testsets used, can be found in [7]. It should be noted that the most common approach to QbyT KWS is training a neural network with an output layer size equal to the number of keywords in the testset extended by special classes to mark the negative examples [8, 9, 10]. Such models cannot be treated as solutions to the custom keyword spotting problem, even if the testset size is large.

Many KWS solutions were tested using the Google Speech Commands (GSC) testset[1]. It was released in two versions: V1 and V2, containing recordings of 30 and 35 words, respectively. Ten words were treated as positive samples (keyword), and the remaining part was used as negative samples (non-keyword). The words were short enough to fit in the recording lasting under one second. The GSC testset contained crowd-sourced recordings from 1881 speakers (V1) and 2618 speakers (V2). Several studies reported their results evaluated on V1 [8, 11, 12, 13], while the others on V2 [8, 13, 14, 15]. The best results so far have been reported by [8] with an accuracy of 98.0 % on V1 and 98.7 % on V2.

Despite its popularity, the GSC testset was not able to evaluate KWS solutions thoroughly due to several reasons: the number of keywords was relatively low, and the negative samples were entirely different from the keywords. All the samples were recorded without background noise and were cut with high precision. Those issues make GSC inadequate for emulating real production conditions.

The Multilingual Spoken Words Corpus (MSWC) [16] contains as many as 23 M keywords split between 50 lan-

[1]P. Warden, "Speech commands: A dataset for limited vocabulary speech recognition," 2018. Available: https://arxiv.org/abs/1804.03209

10.21437/Interspeech.2023-1049

guages. Alignments generated with Montreal Forced Aligner (MFA) [17] were used to extract audio samples. The keywords were selected based on the word length (minimum three letters) and occurrence frequency (minimum five occurrences per language subset). The data was split between *train*, *dev*, and *test* subsets. Unfortunately, this dataset does not contain keywords longer than one word. What is more, it does not provide negative samples to a given keyword, which would allow for a false positive (FP) rate analysis.

MCV contains subsets of short phrases called "Single Word Target Segment", which could be used for KWS evaluation. This approach was applied in [18]. However, those subsets are also very limited, since they contain only up to 14 short keywords (digits and four predefined keywords). Furthermore, there are no negative samples defined for each keyword.

A few papers used modifications of datasets available in the public domain to evaluate proposed solutions. One example is [19], where LibriSpeech was the base for creating the testset. Alignments generated with MFA were used to extract audio samples. The testset contained recordings $0.5\,\mathrm{s}$–$1.5\,\mathrm{s}$ long, including n-grams with $n \leq 4$. This gave 6047 different keywords. Positive samples were combined with two types of negative samples: "confusing" (phonetically similar to the keyword) and "non-confusing" (phonetically dissimilar). Phoneme-level transcriptions and edit distances were used to select both types of negative samples. Even though the description of the testset was fairly precise, the testset itself has not been published.

In [20], a KWS solution based on triplet loss was evaluated; apart from GSC, the test suite included recordings extracted from LibriSpeech. Different testsets were created using 10, 100, 1000, and 10,000 most popular words. Forced alignments were used to extract selected phrases. Since the most popular words were "the", "and", "of", etc., the audio excerpts were relatively short ($0.03\,\mathrm{s}$–$2.8\,\mathrm{s}$). Unfortunately, no information about the sizes of the classes in the testset was given, nor has the testset been published.

The GSC testset, excerpts from LibriSpeech based on alignements generated by MFA, and subsets of MCV were used in [18]. In this case, the testset was built with n-grams ($n \leq 5$) that contained at least 10 characters and had at least 10 occurrences in the train split of the dataset. This gave over $15.2\,\mathrm{k}$ different keywords. Unfortunately, this testset has not been made public, either.

3. Requirements for an optimal custom keyword testset

Analysis of the previous work on KWS shows that the testsets used to evaluate models suffered from several flaws, e.g., a small number of keywords, keywords being very short, only positive samples available, negative samples not challenging, or testsets designed solely for offline evaluation. To create a testset free from these drawbacks, we first defined a set of requirements that, in our opinion, should be followed when building an optimal KWS testset. These requirements are presented below, alongside their justification. Any parametric values were set heuristically during initial experiments.

1. **Keywords should be selected among phrases with the phonetic transcription length p such that $6 \leq p \leq 16$.** Production KWS systems usually have preset requirements for the length of a keyword. Shorter phrases might result in a high FP rate, since usually they are similar to many other phrases, or might be contained in longer phrases. On the other hand, long keywords are impractical for the user. Furthermore, KWS systems are usually deployed on devices with limited computing power, hence the requirement to restrain keyword length.
2. **The testset should contain positive and negative samples for each keyword.** Testing with positive data is not enough, as KWS solutions should also minimize FP rates. Furthermore, the negative samples should be varied and challenging.
3. **Similarity between phrases should be measured with normalized phonetic Levenshtein distance,** which proved to be successful in other studies, such as [19]. It allows us to decide which phrases are similar or different and, consequently, find which phrases are difficult to distinguish.
4. **Keywords should be selected among the words with at least two occurrences.** This requirement assures that in the QbyE KWS task, each keyword will have at least two test cases consisting of pairs of samples in the positive part.
5. **Positive samples for each keyword should, of course, contain phrases having exactly the same phonetic transcriptions,** to measure the true positive (TP) ratio.
6. Negative samples for each keyword should contain:
   - **Recordings containing "similar phrases", i.e., the phonetic distance between the keyword and the tested phrase is in the interval $(0.0, 0.5)$.** These samples should be the most challenging for the model since they would be pronounced similarly to the given keyword.
   - **Recordings containing "different phrases", i.e., the phonetic distance between the keyword and the tested phrase is in the interval $[0.5, \infty)$.** The goal of this type of recording is to ensure a low FP rate on the general types of speech.
7. **The testset should allow for evaluating performance in noisy conditions and in online mode.** Production systems usually work in a challenging environment with different types of background noise. Additionally, keyword spotters most often work in streaming mode. Some solutions assume non-streaming mode, but they receive recordings processed by end-point detectors, which do not work perfectly, either.

4. Proposed MOCKS testset

4.1. Source audio data

Producing and validating new audio data is usually a very expensive and time-consuming process. However, there are numerous datasets that are large and varied enough to select subsets of data for tasks other than speech recognition. For this purpose, we used LibriSpeech and MCV corpora.

4.1.1. LibriSpeech

LibriSpeech has become a standard dataset for speech processing tasks, mostly due to its size ($960\,\mathrm{h}$) as well as the variety of speakers and vocabulary. However, the recordings in this dataset are very particular since they are extracted from English audiobooks read by professional speakers, and each audio sample contains a single sentence. The vocabulary is specific, and the average length of the recording is larger when compared to other datasets ($7.42\,\mathrm{s}$ in *test-clean*, $6.54\,\mathrm{s}$ in *test-other* and $4.94\,\mathrm{s}$–$5.33\,\mathrm{s}$ in MCV, depending on the language). However, its large vocabulary makes LibriSpeech a good candidate for a custom keyword spotting testset when one extracts only short parts of the recordings. In MOCKS, we used data extracted

Table 1: *Properties of MOCKS subsets based on LibriSpeech and MCV*

Property	**en_LS_clean**	**en_LS_other**	**de_MCV**	**en_MCV**	**es_MCV**	**fr_MCV**	**it_MCV**
# Keywords	6883	6918	6581	6534	7694	5540	8062
# Total positive	97924	115334	159890	105126	195246	101764	208626
# Total similar	195360	200350	182305	178747	271258	166338	266288
# Total different	208760	215200	204820	201840	275460	175540	275360
Offline min. len. [s]	0.23	0.35	0.33	0.34	0.24	0.32	0.34
Offline avg. len. [s]	0.80	0.79	0.83	0.89	0.80	0.86	0.90
Offline max. len. [s]	2.54	4.54	5.65	6.20	2.86	2.82	4.06
Online min. len. [s]	1.28	1.31	1.00	0.72	1.11	0.99	1.01
Online avg. len. [s]	2.89	2.87	2.82	2.87	2.75	2.80	3.08
Online max. len. [s]	5.72	6.88	8.48	9.20	7.54	6.60	7.59

from *test-clean* and *test-other* splits. We will refer to the resulting testsets as *en_LS_clean* and *en_LS_other*, respectively.

4.1.2. MCV

MCV is another large dataset available in the public domain. It is based on crowdsourcing and offers a high number of annotated recordings in multiple languages. To prepare our MOCKS testset, we used Version 12.0 of the dataset. We decided to focus on five languages commonly used in Europe: English, German, Spanish, French, and Italian. We will refer to the resulting testsets as *en_MCV*, *de_MCV*, *es_MCV*, *fr_MCV* and *it_MCV*.

4.2. Creation of our testset

We used an internally-developed, rule-based grapheme-to-phoneme (G2P) algorithm to prepare phonetic transcriptions for each sample. Even though numerous phrases contained multiple variants of such transcriptions, we decided to use those which were the most popular to reduce the number of compared phrases. In this case, their popularity was assessed by language experts.

The datasets designed for speech recognition tasks usually contain phrases with phonetic transcriptions that are much longer than the upper bound in our requirements. To increase the number of potential keywords, we decided to use selected fragments of all phrases contained in the processed datasets. We used word-level alignments generated by MFA and models available in the public domain[2] to extract audio data containing keywords. For each keyword, "similar phrases" and "different phrases" were selected so that they would not contain the keyword as a subphrase.

While creating "similar phrases" sets, we decided to use no more than 10 phrases phonetically closest to the given keyword. In case many phrases had the same phonetic distance, random selection was performed. For each phrase, the "different phrases" set contains 10 randomly selected recordings of the types described in the requirements above.

Our testset contains two versions of the audio samples: online and offline. For the offline version, we used MFA-generated timestamps with additional 0.1 s at the beginning and end of the extracted audio sample in order to mitigate the cut-speech effect in the keywords. For the online version, we used MFA-generated timestamps with additional 1 s or so at the beginning and end of the extracted audio sample. The additional amount of audio data might be smaller than 1 s if the keyword appeared at the beginning or end of the recording. If the keyword was surrounded by other words, the amount of additional audio data might be larger than 1 s, since the cut was performed on the nearest aligned timestamp beyond 1 s. The online version of the testset contains timestamps of the keywords.

The final step of the testset preparation procedure consisted of manual checking of the transcriptions in order to exclude obviously incorrect samples.

4.3. MOCKS description and analysis

Below we describe in more detail the contents of MOCKS. In Table 1, several basic statistics on its subsets are presented, such as:

- **# Keywords** – number of keywords,
- **# Total positive/similar/different** – number of pairs in "positive/similar/different phrases" subsets,
- **Offline/Online Min. len. [s]** – minimum keyword recording length in offline and online scenarios,
- **Offline/Online Avg. len. [s]** – average keyword recording length in offline and online scenarios,
- **Offline/Online Max. len. [s]** – maximum keyword recording length in offline and online scenarios.

Using the requirements described in Section 3, we obtained 5000–8000 keywords for each subset. Analysis of the keywords lengths is presented in Figure 1. Most of the keywords are short: in each subset, the keywords with 6 or 7 phonemes constitute approximately half of all the keywords; hence the average length of the recording in the *offline* scenario is under 1 s. The shortest recordings in the *online* scenario also have the length under 1 s, which is caused by the fact that the whole recording for the selected keyword was that short. Since *test-other* and *MCV* already contain data in challenging acoustic conditions, we decided not to mix the data with additional noise.

Both *en_LS_clean* and *en_LS_other* splits are balanced regarding speaker gender distribution. However, the original MCV datasets do not have this property: in most of the considered languages, nearly 60 % of the samples are marked as "male", 8 %–23 % are marked as "female" and less than 2 % of the samples are marked as "other"; there is also a large number of samples with unspecified gender (see Figure 2). To remedy this issue, we randomly drew 2,500 "female" and 2,500 "male" samples for each language to generate keywords.

The data is stored in a 16-bit, single-channel WAV format. 16 kHz sampling rate is used for *en_LS_clean* and *en_LS_other*,

[2] https://mfa-models.readthedocs.io/en/latest/acoustic/index.html

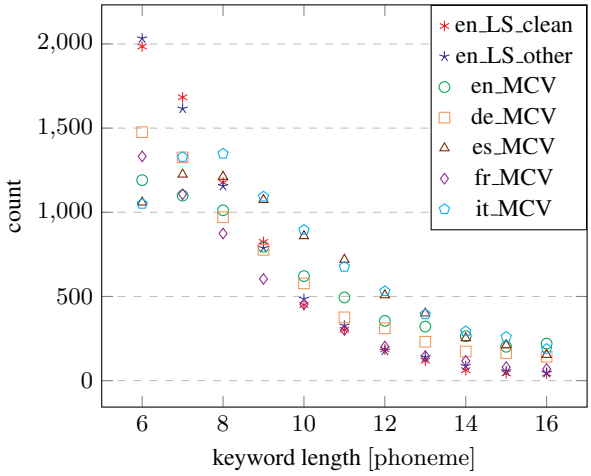

Figure 1: *Distribution of keyword lengths in MOCKS subsets*

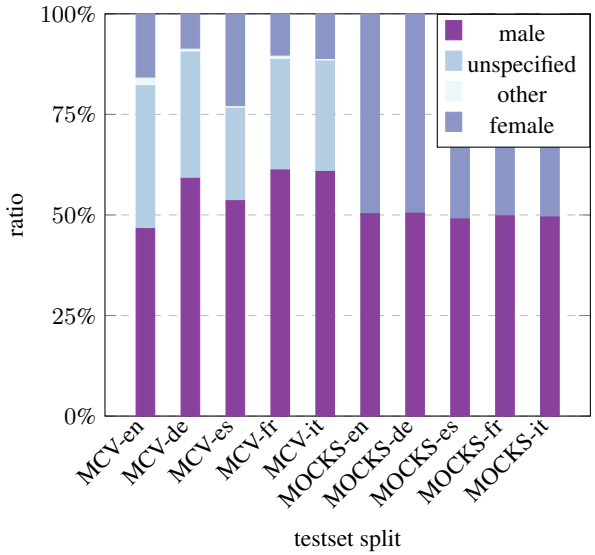

Figure 2: *Gender distribution in MCV and MCV-originated MOCKS subsets*

while 48 kHz for *en_MCV*, *de_MCV*, *es_MCV*, *fr_MCV* and *it_MCV*. This difference is a result of the source datasets' sampling rates. Each testset split contains approximately 500 k test cases, which can be difficult to process, so we also add a subset of MOCKS to allow faster evaluations. Those subsets contain 20 k test cases in each scenario, and each testset split.

5. Initial experiments with MOCKS

To estimate the utility of the proposed testset, we evaluated a baseline model for the QbyE KWS offline task. Our model was based on the solution described in [21]. It consisted of an encoder with 6 bidirectional LSTM layers with 124 cells each, and a linear layer generating 320-dimensional embeddings for audio data. The total number of trainable parameters was 2 M. We pre-trained the encoder model as a part of the Listen, Attend, Spell model [22] on the ASR task using all trainsets from LibriSpeech. Next, we appended the output layer to the pre-trained

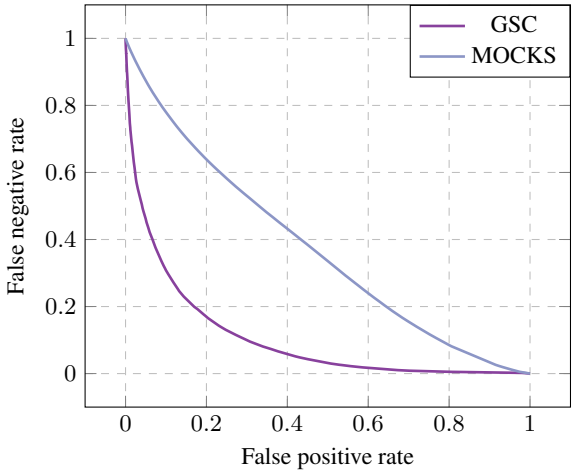

Figure 3: *DET curve for MOCKS compared to GSC testset*

encoder and fine-tuned it for 20 epochs using the contrastive loss function. The dataset in KWS fine-tuning step consisted of recordings generated by an in-house Text-To-Speech solution from approximately 400 English keywords not included in MOCKS. During inference, the Euclidean distance between keyword and test sample embeddings was calculated and compared with a preset threshold. We did not perform any hyperparameter fine-tuning.

In order to compare the results obtained on MOCKS and on previously available testsets, we decided to use GSC, even though it was not prepared for the QbyE task. For each keyword recording in the GSC testset we randomly selected 100 samples with the same phrase, 100 samples with a different phrase, and 100 samples from the "Silence" class. There were approximately 400 k test cases in each of those subsets.

Figure 3 shows the DET curves for GSC and MOCKS testsets. The DET curve for MOCKS was prepared after merging all the subsets of this testset. It should be noted that even though the model performed relatively well on GSC, the results were much worse on MOCKS. This clearly shows room for improvement and confirms how demanding MOCKS is. The estimated equal error rate (EER) value for all MOCKS subsets is 41.64 % with a confidence interval of ±0.15 % (at a 95 % confidence level).

6. Conclusions and future work

This paper introduced the Multilingual Open Custom Keyword Spotting Testset named MOCKS. This testset aims to provide unified means of custom keyword spotting model evaluation and, in this way, to foster research on open vocabulary KWS solutions. We also described a list of requirements that, in our opinion, should be followed while creating such a testset. These requirements can be easily applied to create new testsets in various languages based on other large vocabulary datasets. To create such testsets, two types of additional data are required: phonetic transcriptions and word-level alignment.

In the future, we plan to extend our testset with additional languages contained in MCV and other public datasets. We also plan to work on preparing *train* and *dev* sets based on LibriSpeech and MCV.

7. References

[1] T. Bluche and T. Gisselbrecht, "Predicting Detection Filters for Small Footprint Open-Vocabulary Keyword Spotting," in *Proc. Interspeech 2020*, 2020, pp. 2552–2556.

[2] X. Qin, H. Bu, and M. Li, "Hi-mia: A far-field text-dependent speaker verification database and the baselines," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7609–7613.

[3] B. Kim, M. Lee, J. Lee, Y. Kim, and K. Hwang, "Query-by-example on-device keyword spotting," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 532–538.

[4] J. Hou, Y. Shi, M. Ostendorf, M.-Y. Hwang, and L. Xie, "Region proposal network based small-footprint keyword spotting," *IEEE Signal Processing Letters*, vol. 26, no. 10, pp. 1471–1475, 2019.

[5] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP 2015)*. Brisbane, Australia: IEEE, 2015, pp. 5206–5210.

[6] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," in *International Conference on Language Resources and Evaluation*, 2019.

[7] I. López-Espejo, Z.-H. Tan, J. H. L. Hansen, and J. Jensen, "Deep spoken keyword spotting: An overview," *IEEE Access*, vol. 10, pp. 4169–4199, 2022.

[8] B. Kim, S. Chang, J. Lee, and D. Sung, "Broadcasted Residual Learning for Efficient Keyword Spotting," in *Proc. Interspeech 2021*, 2021, pp. 4538–4542.

[9] D. Seo, H.-S. Oh, and Y. Jung, "Wav2kws: Transfer learning from speech representations for keyword spotting," *IEEE Access*, vol. 9, pp. 80 682–80 691, 2021.

[10] M. Mazumder, C. Banbury, J. Meyer, P. Warden, and V. J. Reddi, "Few-Shot Keyword Spotting in Any Language," in *Proc. Interspeech 2021*, 2021, pp. 4214–4218.

[11] O. Rybakov, N. Kononenko, N. Subrahmanya, M. Visontai, and S. Laurenzo, "Streaming Keyword Spotting on Mobile Devices," in *Proc. Interspeech 2020*, 2020, pp. 2277–2281.

[12] R. Tang, W. Wang, Z. Tu, and J. Lin, "An experimental analysis of the power consumption of convolutional neural networks for keyword spotting," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5479–5483.

[13] A. Berg, M. O'Connor, and M. T. Cruz, "Keyword Transformer: A Self-Attention Model for Keyword Spotting," in *Proc. Interspeech 2021*, 2021, pp. 4249–4253.

[14] I. López-Espejo, Z.-H. Tan, and J. Jensen, "Exploring filterbank learning for keyword spotting," in *2020 28th European Signal Processing Conference (EUSIPCO)*. IEEE, 2021, pp. 331–335.

[15] I. López-Espejo, Z.-H. Tan, and J. Jensen, "A novel loss function and training strategy for noise-robust keyword spotting," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 2254–2266, 2021.

[16] M. Mazumder, S. Chitlangia, C. Banbury, Y. Kang, J. M. Ciro, K. Achorn, D. Galvez, M. Sabini, P. Mattson, D. Kanter *et al.*, "Multilingual spoken words corpus," in *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[17] M. McAuliffe, M. Socolof, S. Mihuc, M. Wagner, and M. Sonderegger, "Montreal Forced Aligner: Trainable Text-Speech Alignment Using Kaldi," in *Proc. Interspeech 2017*, 2017, pp. 498–502.

[18] A. Awasthi, K. Kilgour, and H. Rom, "Teaching Keyword Spotters to Spot New Keywords with Limited Examples," in *Proc. Interspeech 2021*, 2021, pp. 4254–4258.

[19] L. Lugosch, S. Myer, and V. S. Tomar, "Donut: Ctc-based query-by-example keyword spotting," *arXiv preprint arXiv:1811.10736*, 2018.

[20] R. Vygon and N. Mikhaylovskiy, "Learning efficient representations for keyword spotting with triplet loss," in *Speech and Computer: 23rd International Conference, SPECOM 2021, St. Petersburg, Russia, September 27–30, 2021, Proceedings 23*. Springer, 2021, pp. 773–785.

[21] S. Settle and K. Livescu, "Discriminative acoustic word embeddings: Recurrent neural network-based approaches," in *2016 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2016, pp. 503–510.

[22] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2016, pp. 4960–4964.