KWA: A New Method of Calculation and Representation Accuracy for Speech Keyword Spotting in String Results

Nguyen Tuan Anh[1]
School of Electronic and Information Engineering,
South China University of Technology,
Guangzhou 510641, P.R.China

Hoang Thi Kim Dung[2]
Faculty of Civil and Environment,
Thai Nguyen University of Technology,
Thai Nguyen, Vietnam

***Abstract*—This study proposes a new method to measure and represent accuracy for Keyword Spotting (KWS) problem in non-aligned string results. Our approach, called Keyword Spotting Accuracy (KWA), was improved from the Levenshtein Distance algorithm, that used to evaluate the accuracy of the keywords in KWS by measuring the minimum distance between two strings. The main improved algorithm is to show the status of each keyword in training phase for predicted and true labels. In which, representing which words are correct, which ones need to be inserted, substituted or deleted when comparing the prediction labels with true ones during the training phase. In addition, a new method of presenting the multiple keywords in results was proposed to indicate the accuracy of each keyword. This method can display detailed results by keywords, from which, we can obtain the accuracy, distribution, and balance of the keywords in the training dataset by actual speech variance, not by counting keywords in true labels as usual.**

***Keywords*—Speech Keyword Spotting; KWS; keyword accuracy; Keyword Spotting Accuracy (KWA); speech recognition**

I. Introduction

The objective of this study is researching the evaluating methods of the speech keyword spotting (KWS) problem when results in string. The goal of the KWS problem is to detect predefined keywords in a stream of user utterances, usually used as device wake-up words, speech enable for smart devices or find the keywords in video or audio files.

KWS has developed for many years, with significant progress and quality of algorithms. The methods are also very diverse, using Audio only, without labels [3]. Both audio and label are used for supervised learning, from using traditional methods [8], to the basic forms of deep learning [16], and Deep Neural Network Based types are of great interest [28], [30], [2], [16], [15], with different methods of evaluating results, but all of them have not solved the KWS results as a string.

Currently in speech recognition and KWS, it is possible to classify into two categories: classification and regression. KWS is classified into binary and multiple classes in the classification.

The first type, binary classification, is usually a type of wake-up word, applied in electronic products such as smartphones and smart devices. Some companies are using this type such as Apple with “Hey Siri", Google with “Hello Google", Xiaomi with “Xiao Ai Tong Xue". In this type, it usually only has one keyword, the length of the keyword has little variation in speech data. The KWS’s mission is to find out in a utterance that contains or not a keyword, so it is classified into binary classification problem. For example, with google, a user said “OK Google, open gmap", after the phrase “OK google" is detected, a connection will be opened so that the device can communicate directly to a server, and then the server will do the task in the end of the command that converts “open gmap" into text, understand the semantics and transfer the command to the device to serve the user.

The second type, multiple-class classification, the goal of this type is to classify utterances into groups. Such as in game applications, keywords are forward, backward, left, right, up, down, etc. each keyword is a utterance in the data set with the same length. In 2017, Google has created a dataset with a list of these keywords, called Google Speech Command. This dataset contains 35 keywords, each of them has one second long, classified into 36 separate groups [33].

With regression type, a data set consists of utterances, with different lengths, in each utterance that can be contained or not one or more keywords in a given keyword list. True labels are strings, they are not classified into groups, and the position of each word in speech data also unknown. KWS’s task is to check if the keywords are in utterances, if they are, then which keywords. In essence, this problem is similar to the Speech Recognition problem, but with a much smaller set of word as keywords, the remaining words are garbage [6].

To measure results, in the classification type, there are some methods to do, like confusion matrix, including true positive (TP), true negative (TN), false positive (FP), false negative (FN) and measures based on those values [34], in article [32], they used this method to to present the results. Based on these methods, a model based on parameters is evaluated such as true positive rate (TPR), true negative rate (TNR), false positive rate (FPR), false negative rate (FNR), accuracy (ACC), F1 score. With these methods, it is easy to calculate the confusion matrix, but this method can not apply to string results, because when only one character changes, the comparison result is no longer accurate. In the regression type, there are some system assessment measures such as: word error rate (WER), token error rate (TER), character error rate (CER), word accuracy (WACC). Speech recognition (SR) accuracy

measurement is mainly based on word error rate (WER) [36], It is calculated based on the Minimum Edit Distance algorithm, and calculations based on unit of word. WER is an effective tool to compare and evaluate the accuracy of different systems as well as the improvement of a system. In KWS, the concept of Token Error Rate (TER) is also used, instead of using WER, it uses each keyword (possibly containing multiple words) as a unit of calculation. Character Error Rate (CER) is used similarly to WER, but the unit of measurement is based on characters. These methods can evaluate the system accuracy, but if a systems with zeros-resource is developed, we will need more information, such as the number of utterances of each keyword, the accuracy of each keyword, the ratio between accuracy and the number of utterances (because of some languages, like Chinese, there are variation, changing the pronunciation according to the words standing next to each other), if using WER only, it is impossible to know exactly.

There are several methods to evaluate the system, based on the calculation of the correct and incorrect prediction of the predictive labels with the true labels such as Term Weighted Value (TWV), Maximum Term Weighted Value (MTWV) [10]. In paper [4], they used Actual TWV (ATWV), they only consider whether or not the keyword is in the predictive label. In the article [17], they used P@n method to present results of top n keywords. In the article [22], they introduced the DR/FA evaluation method for telephone speech, these methods can evaluate the models, but still evaluate the accuracy of entire keyword set, so the problem of estimating the accuracy of each keyword is still unresolved. it is hard to know how many keywords have correctly predicted, not predicted or missed, when the output of KWS model is a string and when training, only accuracy of entire data set is calculated, by calculating the minimum string distance of predicted labels by true labels. When studying the evaluation method of KWS problem, we found that it is difficult to measure the accuracy of each keyword on predicted results. Because KWS model returns the results as strings, so it is difficult to determine the accuracy in percent of each word. But this analysis is necessary, allowing us to know 11the distribution of each keyword in the data-set, especially with words that have multiple pronouncement ways, mutations and modifications as in Chinese or dialect in other languages, for example, see Fig. 1. The more variation, the more data is needed for a keyword during training. Evaluating a KWS model is to evaluate the accuracy of predicted outputs compared to the true labels in the form of string. This study focuses on solving this problem.

Different from the existing assessment methods, the objective of this study is to provide a method for calculating the accuracy of each keyword in the output sequence of the Regression problem. Proposing a method to display the results on a new chart type so that we can observe the number of keywords in the data set, the number of correct predictive keywords, false predictions and unpredictable, that's also the reason because the name Keyword Accuracy is selected.

II. Theory

Making it easier to compare methods, some theory of representing results for the KWS problem is reviewed. As mentioned above, the existing results representations method can be classified into two categories, classification and regression.

Write	pingyin	Read/say
你好	nǐ hǎo	→ ní hǎo”
我很好	Wǒ hěn hǎo	→ “Wǒ hén hǎo” / “wó hén hǎo”
不爱	Bù ài	→ Bú ài
不变	Bù biàn	→ bú biàn
一共	Yīgòng	→ yí gòng

Figure 1. Chinese characters, when reading and writing differently

TABLE I. TYPICALLY USED ERROR RATES AND THEIR SYNONYMS

Name	Acronym	Formular	Synonyms
False Positive Rate	FPR	$\frac{FP}{FP + TN}$	False Accept Rate (FAR), Fall-out
False Negative Rate	FNR	$\frac{FN}{FN + TP}$	False Reject Rate (FRR), False Alarm Rate
True Positive Rate	TPR	$\frac{TP}{TP + FN}$	True Accept Rate, Sensitivity, recall, Hit Rate
True Negative Rate	TNR	$\frac{TN}{TN + FP}$	True Reject Rate, Detection, Rate, Specificity, Selectivity
Positive Predictive Value	PPV	$\frac{TP}{TP + FP}$	Precision
Accuracy	ACC	$\frac{TP + TN}{TP+TN+FP+FN}$	
F1 score	F_1	$\frac{2TP}{2TP + FP + FN}$	

Classification type is easily calculating results into confusion matrix parameters such as true positive, false positives, false negatives, true negatives. The second type, regression, is a comparison between the predicted string labels and the true labels that currently applied by WER and the result is accuracy over the entire data set. In this study, the regression model is focused for strings predicted results.

The first method, the Confusion matrix and related formulas, aims to evaluate accuracy in binary and multiple class classification. To classify results, with binary classifiers, predictive results is classified into one of the two classes that are real positive cases and real negative cases; With multi-keywords, the results are classified into n*n matrices with n being the number of keywords. In a dataset, the number of real positive cases is called condition positive (P), the number of real negative cases is called condition negative (N). Since then, the predicted results are classified into one of four categories, accurate predictions include true positive (TP) and true negative (TN), incorrect predictions include false positives (FP) and false negatives (FN). From the predicted results, the relevant results is calculated as in Table I, equations obtained from [25], [9], [34], [20]. Finally, we have methods to evaluate results based on those formulas via ROC curves, e.g TPR/FPR [21], Precision/Recall [26], [14], False reject Rate/ False Alarm Rate [7], [29], False Negative Rate/Hourly False Positives [1]

The second method, P@k. In the article [27], the accuracy algorithm was used the formula (1) for evaluating method. The

returned result is the accuracy of top k keywords in the system.

$$P@k = \frac{|\{W_r\} \cap \{kW_p\}|}{|\{kW_p\}|} \tag{1}$$

where W_r is relevant words, kW_p is retrieved words, $P@k$ is a precision measurement. The result returns a number, representing the system's accuracy, for example, $P@6 = 0.617$

The third method, TWV. Term Weighted Value (TWV) is a measurement method of KWS system evaluation, introduced in [10], illustrated by the formula (2-5).

$$P_{miss}(\theta) = 1 - \frac{N_{correct}(\theta)}{N_{true}} \tag{2}$$

$$P_{fa}(\theta) = 1 - \frac{N_{incorrect}(\theta)}{N_{Ninc}} \tag{3}$$

$$TWV(\theta) = 1 - (P_{miss}(\theta) + \beta P_{fa}(\theta)) \tag{4}$$

with:

$$\beta = \frac{E}{V}.(Pr^{-1} - 1) \tag{5}$$

where θ refers to detection threshold, $N_{correct}$, $N_{incorrect}$ refer to the number of keyword correct and incorrect detections, respectively. N_{true} refers to the number of occurrences of keywords in that utterance, N_{Ninc} refers to the number of incorrectly detected keywords in that utterance, $P_{miss}(\theta)$ and $P_{fa}(\theta)$ denote the probability of miss and false alarm, respectively. The cost/value ratio, C/V, is 0.1, thus the value lost by a false alarm is a tenth of the value lost for a miss. The prior probability of a term, Pr, is 10^{-4} [10]. Detection score is greater than or equal to θ, The result of this method returns a number to evaluate the system, such as TWV = 0.1962. Recently some articles, such as [11], also use this measure method to represent their results, and the value also returns a number to evaluate the accuracy of their model. In order to evaluate the number of keywords and their correlations, it is necessary to do more in another way. This method can evaluate the accuracy of the model, but in speech, it does not only simply consider that true label and predicted label contain which keywords but also consider the order in which these words appear. So the WER method is based on the Minimum Edit Distance, which is still used in many speech recognition systems. There are two other methods to calculate accuracy based on TWV method of Actual TWV (ATWV) and Maximum TWV (MTWV). ATWV uses actual decisions to represent the system's ability to predict the optimal operating point given by the TWV scoring metric. MTWV is a TWV value of θ yields the maximum TWV [10]. This method is used by some studies such as [5], [12],

The fourth method, MED. The Levenshtein algorithm [18], [35] used to calculate the Minimum Edit Distance (MED) between two strings. Suppose the two strings given for comparison are s and t, the length of the strings is $|s|$ and $|t|$, minimum edit distance is calculated according to the formula (6) ([18], [35]):

$$MED_{s,t}(i,j) = \begin{cases} \max(i,j) & if\ \min(i,j) = 0 \\ min \begin{cases} MED_{s,t}(i-1,j) + 1 \\ MED_{s,t}(i,j-1) + 1 \\ MED_{s,t}(i-1,j-1) + 1_{s \neq t} \end{cases} & otherwise \end{cases} \tag{6}$$

If $s_i \neq t_j$ then $1_{(s_i \neq t_j)} = 1$ and 0 otherwise, $\mathtt{MED}_{s,t}(i,j)$ is the smallest distance of the first i characters of s compared to the first j characters of t To measure the accuracy of a model, Word Error Rate (WER) is used, calculated according to the formula formula (7) [36].

$$\mathtt{WER}_{s,t} = \frac{S + I + D}{N} = \frac{\mathtt{MED}_{(s,t)}}{N} \tag{7}$$

Where S, I and D represent the number of substitutions, insertions and deletions, N is the number of words in the reference.

In order to evaluate a KWS problem, we have four main methods as mentioned above, but in all of them, there is no one strong enough to calculate the accuracy of each keyword that one or more keywords are inside a string; Displays the balance distribution of each keyword in the data set. That is the motivation for us to carry out this research. Moreover, this study has provided a new way of displaying graphics, thereby fully demonstrating simultaneous information. That is the motivation for this research to be done

III. Propose Method

In this study, we propose an algorithm that calculates the accuracy of the model according to the keyword, with the model output being a string of characters that can have keywords or not, and proposes a new method of representing the results. This one is improved from the Minimum Edit Distance algorithm of Levenshtein for the KWS problem. The output of regression model is a string, to match the multi-lingual problem (like Chinese and Vietnamese, completely different from the structure of words). We introduce an algorithm in equation (8) so called Speech Keyword Accuracy (KWA), to determine the exactly editing position of each keyword, based on the minimum edit distance. To be compatible in multiple languages, each label will be separated into a list of words, in Chinese, separated by each character, in Vietnamese separated by space between words.

$$MED_{s,t}(i,j) = \begin{cases} \begin{cases} i \\ TOC_{1..i,j} = M_{ins} \end{cases} & if\ j = 0 \\ \begin{cases} j \\ TOC_{i,1..j} = M_{del} \end{cases} & if\ i = 0 \\ min \begin{cases} \begin{cases} MED_{s,t}(i-1,j) + 1 \\ TOC_{i,i} = M_{del} \end{cases} \\ \begin{cases} MED_{s,t}(i,j-1) + 1 \\ TOC_{i,j} = M_{inc} \end{cases} \\ \begin{cases} MED_{s,t}(i-1,j-1) + 1 \\ TOC_{i,j} = M_{sub} \end{cases} & if\ s_i \neq t_j \\ \begin{cases} MED_{s,t}(i-1,j-1) + 1 \\ TOC_{i,j} = M_{eq} \end{cases} & if\ s_i = t_j \end{cases} & otherwise \end{cases} \tag{8}$$

In the KWA algorithm in equation (8), the input is provided by two lists s, t and a list output TOC (abbreviation of type of changes), in which each element is equal, substitution, insertion or deletion, denoted by M_{eq}, M_{sub}, M_{inc} and M_{del}, respectively, each of them is a constant number. The result is updated to a global variable, from there, accuracy of each keyword is obtained as in equation (12), the accuracy of the whole model across the dataset as definition in equation (13).

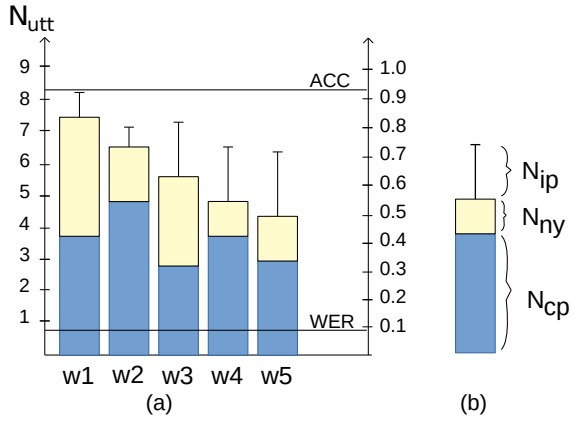

Figure 2. Example of presentation of Speech Keyword Accuracy (KWA) algorithm
N_{utt}: Number of utterances,
w_i (i=1,2...): predefined keywords,
ACC: Model's accuracy,
WER: keyword error rate of model,
N_{ip}: Number of keywords incorrectly predicted (not in true label),
N_{ny}: The number of keywords not yet predicted,
N_{cp}: Number of keywords correctly predicted.

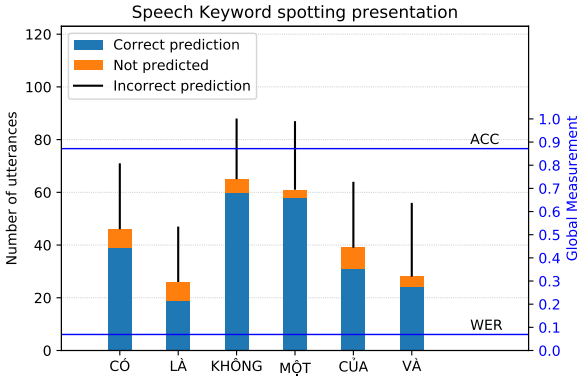

Figure 3. The graph shows the correlation of results between keywords of ViVos dataset

WER based on TOC also observed as in equation (7), where, in each utterance, parameters is calculated as in equation (9-11).

$$S_i = \sum_j (TOC_{i,j} == M_{sub}) \tag{9}$$

$$I_i = \sum_j (TOC_{i,j} == M_{inc}) \tag{10}$$

$$D_i = \sum_j (TOC_{i,j} == M_{del}) \tag{11}$$

or $WER = MED_{s,t}/N$.

This study also propose a method to presenting results in a graph to easily observe the accuracy of each keyword in the keywords set. In Fig. 2, The total number of each keyword occurrences denote as N_{kw}: $N_{kw} = N_{ny} + N_{cp}$. This representation method tells us the overall WER of that system, the number of keywords, the status of each keyword, how many percent each keyword predicted correctly, correlation in terms of number of keywords included in dataset and the number of incorrectly predicted words and not yet predicted. That information can be read along the vertical axis on the left. According to the vertical axis on the right, the results in accuracy as a percentage and WER can be observed, either of which may be missing. During training, incorrectly predicted words can have many reasons, which may be due to lack of data, imbalance in the data set (in classification of images dataset or isolated speech dataset maybe easier to identify than speech recognition dataset). From here, in training process, we will be known that which keywords is needed to prepare more training data so each keyword can be balanced on WER with others. The formula for calculating ACC [23] for each keyword (acc_i) is given in equation (12), and global ACC can be calculate as in (13).

$$acc_i = \frac{N_{cp} - N_{ip}}{N_{cp} + N_{ny}} \tag{12}$$

$$ACC = \frac{1}{N} \sum_{i=0}^{N-1} acc_i \tag{13}$$

where $N_{cp}, N_{ip}N_{ny}$ refer to number of correctly predicted, incorrectly predicted and not predictable, respectively. N denotes as the number of utterances in the dataset. Here, parameters is calculated as equation (14-16)

$$N_{cp}(i) = \sum_j (TOC_{i,j} == M_{eq}) \tag{14}$$

$$N_{ip}(i) = \sum_j (TOC_{i,j} == \{M_{del}|M_{sub}\}) \tag{15}$$

$$N_{ny}(i) = \sum_j (TOC_{i,j} == M_{inc}) \tag{16}$$

IV. Experiments and Results

To do the experiment, we selected two small database sets, representing the low-resources languages, ViVos and THCH-30.

A. Dataset

THCHS-30 corpus. THCHS-30 corpus is an open speech Chinese database [31], publicized in openslr [24], for a total of up to 30 hours for free of reading audios with labels, recorded in a quiet room. This corpus has the characteristics as shown in Table II. To get results for the KWS problem, 10 keywords are selected and implemented by taking 10 words with the highest occurrence frequency in the entire data set to perform the test. After selecting, we have the following keyword list:

KW=[的, 一, 有, 人, 了, 不, 为, 在, 用, 是]
(De, yī, yǒu, rén, le, bù, wèi, zài, yòng, shì)

ViVos corpus. ViVos corpus is a open speech Vietnamese data set [19]. It includes 15 hours of voice recording for Automatic Speech Recognition (ASR) purposes. published by

TABLE II. STATISTICS OF THCHS-30 DATABASE

Dataset	Speaker	Male	Female	Age	Utterance	Duration(hour)
Train	30	8	22	20-50	10893	27:23
Test	10	1	9	19-50	24	6:24

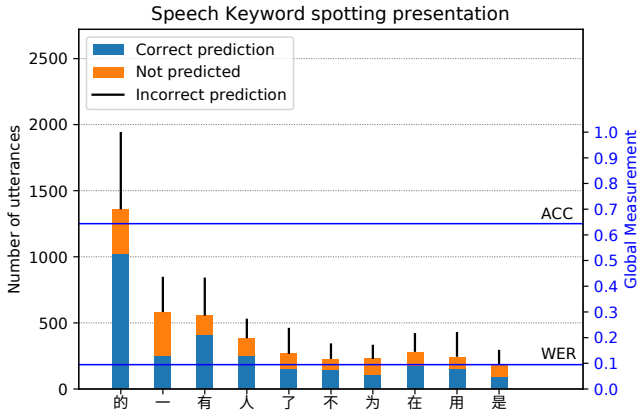

Figure 4. The graph shows the correlation of results between keywords of THCH-30 dataset

AILAB, VNU's computer science laboratory - Hanoi University of Technology. Descriptive characteristics are shown in Table III. The method of selecting keywords is the same as on THCH-30 dataset, and the keyword list has been selected including 6 keywords as:

KW= [Có, Là, Không, Một, Của, Và]

These two sets of data will be used to train with LSTM-CTC model based on [13], outputs of the model and true labels are saved to calculate KWA and display results.

B. Presentation Method

Both ViVos and THCH-30 data sets are trained by LSTM-CTC model, during training, the model is evaluated by CTC loss, based on [13]. CTC loss does not show us how much the accuracy of the model is, but it is possible to evaluate the same model, the same data set, which training session has lower loss, the weight is better. From there the training system can be optimized, to give out the predicted results of the model and combine it with true labels, calculate accuracy according to each keyword and overall accuracy. The formula (12) and (13) are used. The result of this step, is shown on the graphic.

In Figure 3, we can observe, firstly, the number of each keyword is small, and therefore, the difference between the keywords is small, but the percentage is large. Secondly, although the model of accuracy results is quite high, but the percentage of incorrect prediction is also high, and finally, observing WER and accuracy of the system visually, giving us an overview of the model.

TABLE III. STATISTICS OF VIVO CORPUS

Dataset	Speaker	Male	Female	Utterance	Duration(hour)	Unique Syllables
Train	46	22	24	11660	14:55	4617
Test	19	12	7	760	00:45	1692

In Figure 4, it can easily be observed that a huge difference in the number of keywords, the first keyword has approximately twice to sixth times the number of remaining keywords, this leads to difficult for training model to get higher accuracy for the entire set of keywords in the dataset. On the other hand, it is observed that in the second keyword bar, ACC of this keyword has not reached about 50%, while other keywords having higher ACC, thereby giving us a clue to understanding the cause of global ACC is not high.

V. CONCLUSION

We have just described an improved speech keyword spotting accuracy measurement method (KWA) and a new way of presenting results, providing useful information for the training deep learning model. With the KWA method, the accuracy, WER, keyword accuracy, hit/miss in two string label sets, predicted label and true label sets, were evaluated based on improved minimum edit distance algorithm. Besides, the KWA presenting method also provides a figure that we can observe how many keywords correctly, incorrectly predicted and not yet predicted out,the accuracy and WER of the model in a figure. This method helps us understand the balance of keywords in the data set instead of WER or accuracy only. Despite many advantages, KWA still cannot avoid such complex drawbacks. Only string data should be used. In many cases it is not necessary to use an accuracy rating to each keyword. This method can be applied to Speech Recognition problem for almost zero-resource languages and semi-supervised ASR, which will be our future research work.

REFERENCES

[1] A. Abdulkader, K. Nassar, M. Mahmoud, D. Galvez, and C. Patil. Multiple-instance, cascaded classification for keyword spotting in narrow-band audio. *ArXiv*, abs/1711.08058, 2017.

[2] M. A. Al- Rababah, A. Al-Marghilani, and A. A. Hamarshi. Automatic detection technique for speech recognition based on neural networks inter-disciplinary. *International Journal of Advanced Computer Science and Applications*, 9(3):179–184, 2018.

[3] M. Awaid, A. H., and S. A. Audio Search Based on Keyword Spotting in Arabic Language. *International Journal of Advanced Computer Science and Applications*, 5(2):128–133, 2014.

[4] Y. Bai, J. Yi, H. Ni, Z. Wen, B. Liu, Y. Li, and J. Tao. End-to-end keywords spotting based on connectionist temporal classification for mandarin. In *2016 10th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, pages 1–5. IEEE, 2016.

[5] Y. Bai, J. Yi, H. Ni, Z. Wen, B. Liu, Y. Li, and J. Tao. End-to-end keywords spotting based on connectionist temporal classification for Mandarin. In *Proceedings of 2016 10th International Symposium on Chinese Spoken Language Processing, ISCSLP 2016*, 2017.

[6] E. Chandra and K. A. Senthildevi. Keyword Spotting: An Audio Mining Technique in Speech Processing – A Survey. *IOSR Journal of VLSI and Signal Processing Ver. II*, 5(4):22–27, 2016.

[7] G. Chen, C. Parada, and G. Heigold. Small-footprint keyword spotting using deep neural networks. *Acoustics, Speech and Signal ...*, i:1–5, 2014.

[8] H. F. C. Chuctaya, R. N. M. Mercado, and J. J. G. Gaona. Isolated Automatic Speech recognition of Quechua numbers using MFCC, DTW and KNN. *International Journal of Advanced Computer Science and Applications*, 9(10):24–29, 2018.

[9] T. Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

[10] J. G. Fiscus, J. Ajot, J. S. Garofolo, and G. Doddington. Results of the 2006 spoken term detection evaluation. In *Proc. sigir*, volume 7, pages 51–57, 2007.

[11] M. J. Gales, K. M. Knill, A. Ragni, and S. P. Rath. Speech recognition and keyword spotting for low-resource languages: Babel project research at cued. In *Spoken Language Technologies for Under-Resourced Languages*, 2014.

[12] M. J. F. Gales, K. M. . Knill, A. Ragni, and S. P. . Rath. Speech recognition and keyword spotting for low resource languages: Babel project research at CUED. In *Spoken Language Technologies for Under-Resourced Languages (SLTU)*, number May, pages 14–16, 2014.

[13] A. Graves and N. Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *International conference on machine learning*, pages 1764–1772, 2014.

[14] Y. Huang and W. Y. Wang. Deep Residual Learning for Weakly-Supervised Relation Extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1803–1807, 2017.

[15] P. D. Hung, T. M. Giang, L. H. Nam, and P. M. Duong. Vietnamese speech command recognition using Recurrent Neural Networks. *International Journal of Advanced Computer Science and Applications*, 10(7):194–201, 2019.

[16] M. K, I. A., and G. Onwodi. Neural Network Based Hausa Language Speech Recognition. *International Journal of Advanced Research in Artificial Intelligence*, 1(2):39–44, 2012.

[17] H. Kamper, G. Shakhnarovich, and K. Livescu. Semantic keyword spotting by learning from images and speech. *arXiv preprint arXiv:1710.01949*, 2017.

[18] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710, 1966.

[19] H.-T. Luong and H.-Q. Vu. A non-expert kaldi recipe for vietnamese speech recognition system. Technical report, 2016.

[20] S. Marcel, M. S. Nixon, and S. Z. Li, editors. *Handbook of Biometric Anti-Spoofing*. Advances in Computer Vision and Pattern Recognition. Springer London, London, 2014.

[21] R. Menon, H. Kamper, J. Quinn, and T. Niesler. Fast ASR-free and almost zero-resource keyword spotting using DTW and CNNs for humanitarian monitoring. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, volume 2018-Septe, pages 2608–2612, 2018.

[22] J. Nouza and J. Silovsky. Fast keyword spotting in telephone speech. *Radioengineering*, 18(4):665–670, 2009.

[23] A. Ogawa, T. Hori, A. Nakamura, A. Ogawa, T. Hori, and A. Nakamura. Estimating speech recognition accuracy based on error type classification. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 24(12):2400–2413, 2016.

[24] Open Speech and Language Resources. Thchs-30. http://www.openslr.org/18. [Online; accessed 31-March-2019].

[25] D. M. W. Powers. Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. *Journal of Machine Learning Technologies*, 2007.

[26] I. Pratikakis, K. Zagoris, B. Gatos, G. Louloudis, and N. Stamatopoulos. ICFHR 2014 Competition on Handwritten Keyword Spotting (H-KWS 2014). In *Proceedings of International Conference on Frontiers in Handwriting Recognition, ICFHR*, volume 2014-Decem, pages 814–819, 2014.

[27] I. Pratikakis, K. Zagoris, B. Gatos, G. Louloudis, and N. Stamatopoulos. Icfhr 2014 competition on handwritten keyword spotting (h-kws 2014). In *2014 14th International Conference on Frontiers in Handwriting Recognition*, pages 814–819. IEEE, 2014.

[28] J. Ren and M. Liu. An Automatic Dysarthric Speech Recognition Approach using Deep Neural Networks. *International Journal of Advanced Computer Science and Applications*, 8(12):48–52, 2017.

[29] T. N. Sainath and C. Parada. Convolutional Neural Networks for Small-footprint Keyword Spotting. *Proceedings INTERSPEECH*, pages 1478–1482, 2015.

[30] M. Walid, B. Souha, and C. Adnen. Speech recognition system based on discrete wave atoms transform partial noisy environment. *International Journal of Advanced Computer Science and Applications*, 10(5):466–472, 2019.

[31] D. Wang and X. Zhang. Thchs-30: A free chinese speech corpus. *arXiv preprint arXiv:1512.01882*, 2015.

[32] Z. Wang, X. Li, and J. Zhou. Small-footprint keyword spotting using deep neural network and connectionist temporal classifier. *arXiv preprint arXiv:1709.03665*, 2017.

[33] P. Warden. Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition. *arXiv preprint arXiv:1804.03209*, 2018.

[34] Wikipedia contributors. Confusion matrix — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Confusion_matrix&oldid=881721342, 2019. [Online; accessed 31-March-2019].

[35] Wikipedia contributors. Levenshtein distance — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Levenshtein_distance&oldid=887999285, 2019. [Online; accessed 28-March-2019].

[36] Wikipedia contributors. Word error rate — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Word_error_rate&oldid=888037079, 2019. [Online; accessed 31-March-2019].