CONFIDENCE ESTIMATION FOR LATTICE-BASED AND LATTICE-FREE AUTOMATIC SPEECH RECOGNITION

Alexandru CARANICA[1], Dan ONEAȚĂ[2], Horia CUCU[3], Corneliu BURILEANU[4]

In this paper we study methods for estimating the confidence of each word in a transcription generated by an automatic speech recognition (ASR) system. A common benefit of confidence scoring is the increased robustness of the system, for example, by discarding uncertain transcriptions for applications that require high accuracy. The confidence scoring methods depend on the type of ASR and there are two main classes of such systems: the classical, HMM lattice-based system and the more recent, end-to-end, lattice-free approach. While confidence estimation in lattice-based systems has been extensively studied in the past, similar methods have just started to be explored in the end-to-end case. Here we extend our previous work on confidence scoring for end-to-end systems, by (i) carrying a detailed comparison with the classical approaches and (ii) evaluating on several new datasets in a different language, Romanian. Our results show that the confidence scores for end-to-end systems are very promising, obtaining performance comparable or better than the classical approaches, with the added benefit of being conceptually simpler and more integrated.

Keywords: Automatic speech recognition, confidence scoring, end-to-end deep learning, Romanian language

1. Introduction

Automatic speech recognition (ASR) allows us to interact with the systems and devices around us, in an easy and natural way, through voice commands. Moreover, in recent years, a multitude of downstream applications built on this technology have been developed, ranging from automatic subtitling systems and chatbots to call-center transcription services. As such, research in this domain has led to the constant improvement of performance over time, as state-of-the-art systems, based on sophisticated statistical models and deep learning, reached impressive levels of accuracy.

However, state-of-the-art systems are not perfect and often produce transcripts that contain recognition errors [1]. It is important to have a way of

[1] Postdoctoral researcher, University POLITEHNICA of Bucharest, alexandru.caranica@speed.pub.ro
[2] Postdoctoral researcher, University POLITEHNICA of Bucharest, dan.oneata@speed.pub.ro
[3] Associate professor, University POLITEHNICA of Bucharest, horia.cucu@upb.ro
[4] Professor, University POLITEHNICA of Bucharest, corneliu.burileanu@upb.ro

measuring how confident we can be that the system output is correct. As such, the degree of confidence in an ASR system can be measured using confidence scores.

In recent years, as end-to-end systems are gaining traction, we can also see a shift in "perspective" in the ASR domain. Historically, confidence scoring for ASR has been addressed mostly with lattice-based ASR systems [2][3][4][5], while just recently machine learning research has started to focus on confidence measures for lattice-free ASR systems [6]. These end-to-end models for ASR can match the performance of a classical system and can also bring additional benefits to the table, by being conceptually simple and allowing unified GPU-assisted training [7][8][9], on modern hardware.

However, there is surprisingly little work on confidence estimation for end-to-end ASR systems, as most of the ongoing research on confidence estimation is being carried on computer vision tasks, like image classification or segmentation. There are two challenges that are particular to our task: (i) ASR systems are structured models (sequences-to-sequence models), as opposed to traditional recognition networks used in image classification; (ii) since we need to estimate confidence in an auto-regressive context, the sequential nature of the output imposes a decoding step, which complicates not only the prediction but also the confidence scoring algorithm.

In a previous work we proposed one of the first word-level confidence estimation methods for end-to-end ASR systems [10]. We empirically validated our method on English speech and compared various features, word-level aggregation methods and techniques for further improving the posterior probabilities - the main ingredient in estimating the confidence scores.

As end-to-end systems gradually replace classical, lattice-based, ASR systems, it is important to establish a baseline for both architectures. As such, our main contributions in this paper are the following: (i) we evaluate our previously proposed confidence estimation method on a new language, (ii) we propose a methodology for comparing our state-of-the-art confidence estimation method for lattice-based ASR systems [11] with the proposed method in [10], which uses an end-to-end ASR.

2. Related work

The task of obtaining a good estimate of the recognizer's confidence is challenging and decades of research work has been devoted to improving this metric [3]. Most prior work on confidence scoring for ASR targets "classical systems", based on the HMM-GMM paradigm. These methods first extract a set of features from the decoding lattice, acoustic or language model, and then train a classifier to predict whether the transcription is correct or not. These existing methods for lattice-based confidence scoring are classified in [2].

Confidence estimation for end-to-end ASR systems was firstly addressed by Malinin and Gales [12]. They estimate token-level and sentence-level confidence scores using ASR ensembles in order to generate a probability density over the predictions. In [13], authors present another novel method, which uses internal neural features of a frozen ASR model to train an independent neural network to predict a softmax temperature value. Resulting softmax values corresponding to predicted tokens are then used to constitute a more reliable confidence measure. Another method of obtaining probability densities over the predictions, which we also employed in our recent work [10], involves performing the inference several times using a single ASR system with drop-out layers [14].

To the best of our knowledge, the first word-level confidence estimation method for end-to-end ASR was recently proposed in [10]. In this work, we adapted several state-of-the-art uncertainty estimation methods to the end-to-end ASR pipeline, we proposed and evaluated aggregation techniques to obtain word-level confidence estimates and we performed a thorough evaluation on multiple speech benchmark datasets.

A second work on word-level confidence estimation for end-to-end ASR was published in [15]. In this paper the authors proposed an extension to a light-weight confidence estimation module for end-to-end ASR models, to directly estimate word-level confidence with self-attention and deliberation, by learning from the full acoustic and linguistic context of subword sequence and multiple hypotheses.

3. Methodology

In this section we describe the methodology we used to compare two architecturally different confidence scoring methods. In addition, in subsections 3.1 and 3.2 we briefly describe the two confidence estimation methods.

Comparison of confidence estimation methods for ASR is a difficult task, mainly because these methods are tightly connected to the underlying ASR system. Most of the time, confidence estimation methods use features extracted from the different stages of an ASR decoder. The problem is that if one ASR system is more accurate, spotting confidently its errors becomes more difficult than in the case of the second, less accurate ASR system.

Our goal in this paper is to compare objectively two confidence estimation methods which are tightly linked to different underlying ASR systems: (i) Minimum Bayes Risk (MBR) confidence scoring method applied on top of a lattice-based ASR system and (ii) end-to-end (E2E) [5] confidence scoring method applied on features extracted from a lattice-free ASR system.

The classical, lattice-based ASR system implemented using Kaldi [16] has the ability to decode input speech using best-path and Minimum Bayes Risk

decoding methods. As presented in subsection 3.2, the second decoding method is also suitable for generating high-quality word-level confidence scores.

The E2E confidence scoring method we recently proposed in [10] is based on token-level posterior probabilities extracted during the speech decoding process of an end-to-end ASR system, implemented using ESPnet toolkit [17].

Our aim is to decouple the confidence estimation methods from the underlying ASR systems in order to be able to compare solely the confidence estimation methods, regardless of the accuracy of the two ASR systems.

To achieve this goal we propose the methodology presented in Figure 1.

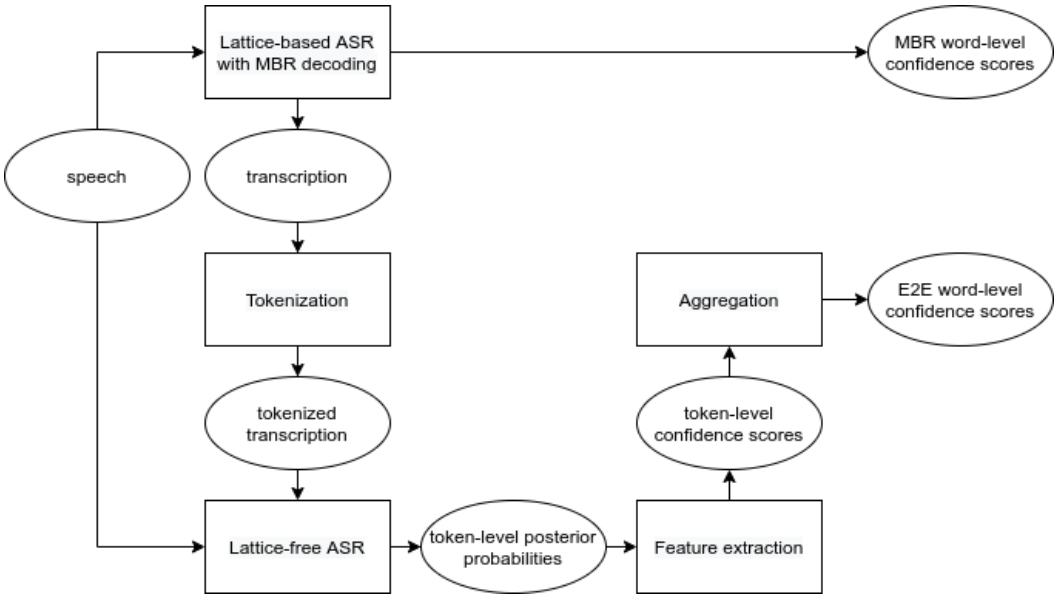

Fig. 1. Confidence estimation comparison methodology.

The MBR word-level confidence scores are obtained using the usual procedure: the input speech is transcribed using the better, lattice-based ASR system, with MBR decoding. E2E word-level confidence scores are generated using the method proposed in [10] with the following adaptation. The input speech is passed through the lattice-free ASR in order to obtain token-level posterior probabilities. However, the process of generating the posterior probabilities for token k requires information regarding the most probable tokens at times $0,1,\ldots,k-1$. While the lattice-free ASR also generates tokens at times $0,1,\ldots,k-1$, these are less accurate than those generated by the lattice-based ASR. At this point, in order to decouple the confidence estimation evaluation from the underlying ASR system, we use the tokenized transcription generated by the lattice-based ASR. Consequently, the token-level posterior probabilities are

generated using the deep neural model of the lattice-free ASR and the previous tokens proposed by the lattice-based ASR. Going further, we follow exactly the steps proposed in the E2E confidence estimation method: feature extraction and aggregation of token-level confidence scores to finally obtain word-level confidence scores.

3.1. E2E confidence estimation method

In this subsection we briefly present the confidence estimation method recently proposed [10].

To avoid the constraints of using a fixed word dictionary and to allow predictions of new words, end-to-end ASR systems typically use subword tokens at the output. But given that tokens lack semantics, for many downstream applications, we were interested in estimating confidence at word level, and not tokens. To this end, we explored methods of aggregating token-level uncertainty measures into larger units, corresponding to words.

Using the posterior probabilities at each point in time p_k, we extract features to represent the confidence score of each token $s_k{}^{(t)}$. In a second step, we aggregate token-level scores into word-level trust scores $s_j{}^{(w)}$, based on the tokens that belong to each word. Figure 2 offers a graphical representation of the proposed pipeline.

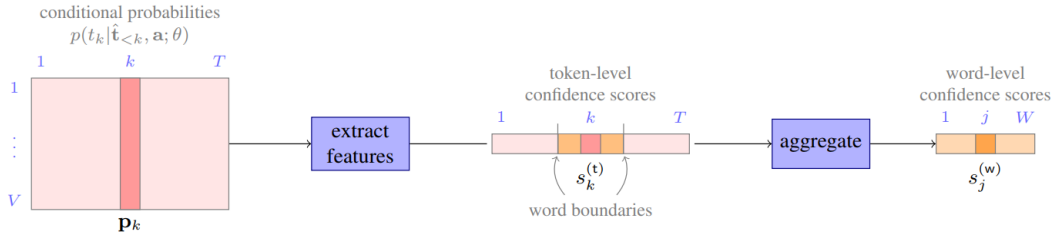

Fig. 2. General schematic representation of the proposed method for estimating confidence scores. Based on an end-to-end speech recognition system (ASR), we obtain p_k probabilities for each token *k*, conditioned by a utterance *a* and the previously predicted tokens $\hat{t}_{<k}$. Based on these probabilities, we extract reliable scores at token level $s^{(t)}$, which we then aggregate to obtain scores at word level $s^{(w)}$. The size of the token vocabulary is denoted by *V*, the number of tokens is denoted by *T* and the number of words by *W*.

Feature extraction. To measure token-level confidence we use three types of features, as shown below. In order to compute these features we use the prediction p (a vector of V elements over the vocabulary of tokens) provided by the lattice-free ASR and the token index k predicted by the lattice-based ASR (i.e., the ASR used to perform the decoding) as the most probable token.

- *Logarithm of the probability* is a direct way of using the prediction p provided by the lattice-free ASR to obtain a confidence measure. *Log-*

proba best is one version of this proposed feature. In this case, we select as a feature for confidence estimation the largest value in the vector, which corresponds to the most probable prediction at a given time step t: $s^{(t)} = log\ max\ p$. This method has been observed empirically to yield a strong baseline for error classification tasks and detection of out-of-distribution samples [18].

- In the second approach of using the logarithm of the probability as a confidence feature, we take into account the token index k predicted by the lattice-based ASR (i.e., the ASR used to perform the decoding) as the most probable token. We call this feature *log-proba pred* because it represents the logarithm of the probability computed by the lattice-free ASR for the token predicted by the decoding ASR (i.e., $s^{(t)} = p_k$). To be more specific, *log-proba pred* is the log-proba indicated by the decoding ASR, *log-proba best* is the maximum value in p. In our previous paper, [10] we have employed only *log-proba best*, while the experiments based on *log-proba pred* are a novel contribution of the current work. We believe the main advantage of *log-proba pred* is that offers a better comparison between the two confidence scoring methods evaluated in this paper. The confidence scoring method based on the lattice-free ASR is, in this case, using more information from the lattice-based ASR decoding.
- *Negative entropy (neg-entropy)* is calculated over tokens vocabulary at each point in time, ie $s^{(t)} = p^T log\ p$. A large entropy means a large uncertainty or, conversely, a large negative entropy implies a reliable prediction.

Aggregation. To obtain word-level characteristics from token-level ones, we experiment with three types of aggregation functions: sum, average, minimum. Summing more tokens will lead to lower values and therefore lower confidence scores. This behavior may be desirable because longer words are more likely to be erroneous. Also, when we sum the probabilities logarithm, we get a word level score corresponding to the log probability of the whole sequence. The use of aggregation with the minimum function is justified by the fact that we might want a low confidence if at least one of the tokens has a low confidence.

In order to make token-level probabilities more reliable, we employed the *dropout* technique. Our assumption is that by improving the probabilities at the token level, we will also get better word-level confidence scores. *Dropout* [19] is a technique that masks random parts of activations in a network, making the network less prone to overfitting.

3.2. MBR confidence estimation

Minimum Bayes Risk decoding aims to find the candidate hypothesis that has the least expected loss under the probability model [20], by selecting the hypothesis that minimizes the expected error during classification. Thus, it directly incorporates the loss function into the decision criterion. MBR decoding is generally implemented by re-ranking an *N*-best list of translations produced by a first-pass decoder.

Our main ASR system is built on top of Kaldi toolkit. To obtain reliable scores in Kaldi, it is necessary to use the Minimum Bayes Risk decoding described above. More specifically, the *lattice-to-ctm-conf* script with the *--decode-mbr true*. Unlike the standard decoding method (based on the *lattice-best-path* script), this method obtains the *w** transcript by optimizing a function of the following type:

$$w^* = argmin_w \sum_{w'} \quad p(w'|x)\, L(w, w'), \tag{1}$$

where *p(*w|x*)* indicates the probability of a word sequence *w* given the audio signal *x*, and *L(*w,w'*)* indicates the Levenshtein distance between two word sequences.

4. Experimental setup

In this section we briefly describe the ASR systems involved in confidence estimation and the evaluation metrics used for comparing the confidence estimation methods.

4.1. ASR systems

The lattice-based ASR system [11] we use further is implemented in Kaldi. The acoustic model is a Hidden Markov Model (HMM) with a Time Delay Neural Network (TDNN) output. It is trained on features such as mel-frequency cepstral coefficients and i-vectors. For decoding it uses a trigram language model. As opposed to the experiments presented in [11], in this paper we do not use language rescoring.

The end-to-end ASR system (lattice-free) is based on ESPnet. It uses both the Connectionist Temporal Classification (CTC) loss and the Transformer based encoder-decoder network (self-attention encoder-decoder). The latter uses self-attention to align between the acoustic frames and the recognized tokens, while the CTC one uses Markov hypotheses to effectively solve sequential problems through dynamic programming. The end-to-end ASR also uses a Transformer language model with a dictionary of 1,000 tokens.

The systems were trained on Romanian speech and text datasets. The acoustic model of the lattice-based ASR was trained on the 225 hrs of speech in RSC-train [21] and SSC-train datasets (see [11] for further details). The Transformer of the lattice-free ASR was trained on slightly more data: we added 292 hrs of speech with automatically generated transcriptions [22]. Finally, the language models of both systems were trained on a text dataset of 352M running words comprising news, interviews and talk shows transcriptions.

The two ASR systems and the confidence estimation methods were evaluated on the RSC-eval [21] and the SSC-eval [11] datasets. The performance of the two ASR systems is presented in Table 1. The lattice-based ASR was evaluated using the two decoding methods (best-path and MBR) in order to make sure that MBR decoding is competitive.

Table 1

Evaluation of the lattice-based and the lattice-free ASR systems. Results are expressed in terms of the word error rate (WER).

ASR system	Decoding	RSC-eval	SSC-eval
Lattice-based	best path	3.5%	19.7%
	MBR	3.5%	19.6%
Lattice-free	n/a	3.4%	15.3%

4.2. Evaluation metrics

Ideally, we want the confidence score to be correlated with the correctness of the transcription, that is, correct words should have a large confidence score, while incorrect ones, low score. Following our previous work [10], we employ metrics that are generally used for evaluating binary classifiers, but which have the discrimination threshold varied. More precisely, we measure the area under *precision-recall curve* (AUPR) and the *area under receiver operating characteristic curve* (AUROC).

Depending on what we want to focus on (correctly or erroneously transcribed words) for AUPR we obtain different variants: if we are interested in detecting erroneously transcribed words, we will treat the errors as the positive class; on the other hand if we are interested in the correctly transcribed words, we will treat the latter as the positive class. Hence, for AUPR we use two variants AUPRe (when errors are treated as positives) and AUPRs (when correct words are treated as positives). For AUROC the same value is obtained for either choice, so there is no need to make this distinction.

5. Results

This section presents our experimental results. We start with a presentation of calibration results for our confidence estimation method in subsection 5.1, then proceed with a comparison between the classical, lattice-based approach and the proposed lattice-free confidence estimation method in subsection 5.2. In the last subsection we compare the confidence estimation methods in terms of their capability of selecting correctly annotated text for a dataset of untranscribed speech.

5.1 Calibration of proposed confidence estimation method

We compare various features and aggregation techniques for the proposed confidence estimation method. Tables 2 and 3 below present the results on all combinations of feature types and aggregation methods for the eval datasets.

We observe that log probability features work best across all combinations of aggregations and features, for both datasets. The newly introduced *log-proba pred* features yield better results than the *log-proba best* features used in our initial work [10].

In terms of the aggregation method, we obtained mixed results. For the RSC-eval dataset the min aggregation works better for all feature types, while for the SSC-eval dataset, the sum aggregation can sometimes be preferable.

Table 2

Evaluation of confidence scoring methods for all combinations of features and aggregations, on the RSC-eval dataset. For all reported evaluation metrics, larger values are better.

		RSC-eval (3.4% WER)		
feat.	**agg.**	**AU-PR error ↑**	**AU-PR sucs ↑**	**AU-ROC ↑**
log-proba best	sum	9.40%	98.36%	68.38%
log-proba best	min	**11.70**%	**98.46**%	**72.34**%
log-proba best	avg	11.27%	98.23%	68.33%
log-proba pred	sum	33.59%	99.07%	82.43%
log-proba pred	min	**33.90**%	**99.12**%	**84.25**%
log-proba pred	avg	33.23%	98.96%	81.91%
neg-entropy	sum	**11.20**%	98.22%	68.23%
neg-entropy	min	10.84%	**98.40**%	**71.45**%
neg-entropy	avg	9.84%	98.12%	66.51%

Table 3

Evaluation of the confidence scoring methods on combinations of features and aggregations, on the SSC-eval dataset. For all reported evaluation metrics, larger values are better.

		SSC-eval1 (15.3% WER)		
caract.	**agg.**	**AU-PR error ↑**	**AU-PR sucs ↑**	**AU-ROC ↑**
log-proba best	sum	36.81%	93.86%	74.72%
log-proba best	min	41.34%	**94.47**%	**78.75**%
log-proba best	avg	**41.80**%	94.17%	77.85%
log-proba pred	sum	**58.84**%	96.63%	86.39%
log-proba pred	min	57.27%	**96.69**%	**87.12**%
log-proba pred	avg	57.89%	96.44%	86.40%
neg-entropy	sum	**41.65**%	**94.17**%	**77.83**%
neg-entropy	min	39.90%	94.14%	77.64%
neg-entropy	avg	40.39%	93.79%	76.44%

Table 4

Evaluation of confidence scoring methods on the RSC-eval dataset using improved probabilities based on the dropout technique, with $N = 64$, for all combinations of features, aggregations. For all reported evaluation metrics, larger values are better.

			RSC-eval (3.4% WER)		
feature	aggregation	N dropout	AU-PR error ↑	AU-PR sucs ↑	AU-ROC ↑
log-proba best	sum	dropout-64	10.88%	98.76%	72.66%
log-proba best	min	dropout-64	**13.26**%	**98.95**%	**77.50**%
log-proba best	avg	dropout-64	12.87%	98.86%	76.12%
log-proba pred	sum	dropout-64	35.22%	99.27%	83.79%
log-proba pred	min	dropout-64	**37.20**%	**99.36**%	**86.18**%
log-proba pred	avg	dropout-64	35.14%	99.29%	85.19%
neg-entropy	sum	dropout-64	6.25%	98.54%	67.21%

neg-entropy	min	dropout-64	**11.11**%	**98.86**%	**75.85**%
neg-entropy	avg	dropout-64	10.27%	98.72%	73.73%

Table 5

Evaluation of confidence scoring methods on the SSC-eval dataset using improved probabilities based on the dropout technique, with $N = 64$, for all combinations of features, aggregations. For all reported evaluation metrics, larger values are better.

			SSC-eval (15.3% WER)		
feature	aggregation	N dropout	AU-PR error ↑	AU-PR sucs ↑	AU-ROC ↑
log-proba best	sum	dropout-64	38.25%	95.15%	78.32%
log-proba best	min	dropout-64	42.86%	**95.77**%	**81.96**%
log-proba best	avg	dropout-64	**44.22**%	95.67%	81.92%
log-proba pred	sum	dropout-64	58.75%	97.16%	87.06%
log-proba pred	min	dropout-64	**58.83**%	**97.45**%	**88.24**%
log-proba pred	avg	dropout-64	59.00%	97.23%	88.04%
neg-entropy	sum	dropout-64	26.58%	93.95%	71.62%
neg-entropy	min	dropout-64	41.40%	**95.46**%	**81.03**%
neg-entropy	avg	dropout-64	**42.36**%	95.32%	80.82%

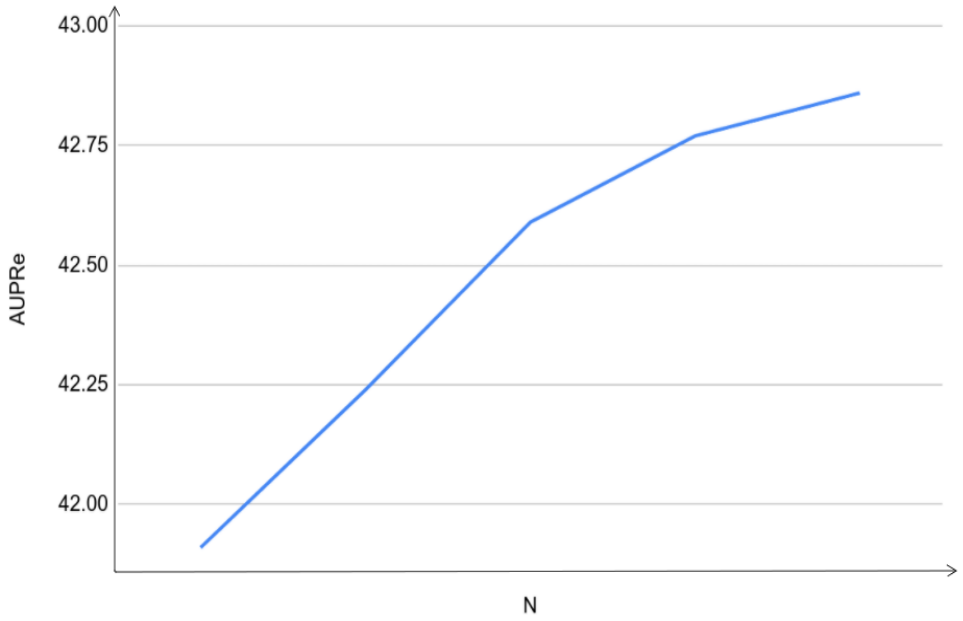

Fig. 3. AU-PR error performance as a function of the number of dropout runs, $N \in \{4,8,16,32,64\}$, on the SSC-eval dataset, using *log-proba best* features and min aggregation. Without dropout, the same combination of features and aggregation yield a performance of 41.34%, see Table 3, row 2.

We have also benchmarked the confidence scoring methods based on improved probabilities, obtained from the dropout technique, which averages over multiple independent predictions (in our case $N = 64$ predictions). Tables 4 and 5 present these results. In Figure 2 we plot the AU-PR error (AUPRe) performance as a function of the number of dropout runs on the SSC-eval dataset, using log-proba features and min aggregation.

We observe that *log-proba* features benefit from dropout more than *neg-entropy* features and offer the best results. Figure 3 shows that the dropout performance improves with the number of runs, then starts to plateau around the value of $N = 64$.

As a result of the calibration experiments performed and presented above, we can conclude that the E2E confidence estimation method works best if we use *log-proba pred* features, min aggregation and dropout with N=64 predictions.

5.2 Comparison with MBR confidence estimation method

In this subsection we compare the calibrated E2E confidence estimation method with the state-of-the-art method available for lattice-based ASR systems using MBR decoding. The experimental results are presented in table 6, below.

Table 6

Comparisons of confidence scores based on three metrics, on RSC and SSC datasets, for the best performing lattice-based system and lattice-free system. For all three metrics, higher values represent better results.

Dataset	Confidence estimation method	AU-PR error ↑	AU-PR sucs ↑	AU-ROC ↑
RSC-eval	MBR	**38.28**%	**99.73**%	**91.40**%
	E2E	37.20%	99.36%	86.18%
SSC-eval	MBR	54.59%	**97.48**%	84.64%
	E2E	**58.83**%	97.45%	**88.24**%

Table 6 shows that the MBR confidence estimation method systematically obtained better results on read speech (RSC-eval dataset), while the proposed E2E method manages to overcome it for spontaneous speech (SSC-eval). We speculate that this happens because the lattice-free ASR system used for extracting features for E2E is significantly better than the lattice-based ASR system (see table 2).

5.3 Selecting automatically generated transcripts for data augmentation

The main idea of this automatic annotation method is to use a high-performance ASR system to produce transcripts for an unannotated, new corpus. The transcribed words that receive high confidence scores will be marked as correct. Finally, the selected transcripts and the corresponding speech segments are used to form a new annotated speech corpus, that can be further used to retrain ASR systems. In order for this method to work, it is essential to use reliable confidence scoring methods that can be used to separate accurately correctly transcribed words from incorrectly transcribed words.

As such, in this section we compare both systems in terms of annotated corpus selection, in order to see if the new confidence estimation method is better for this downstream task.

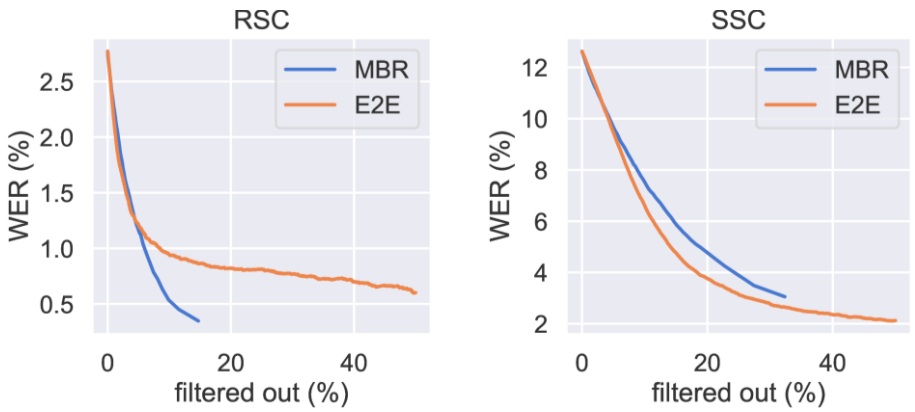

Fig. 4. Word error rate (WER) as a function of the percentage of filtered out data for the two evaluated confidence scoring methods (MBR and E2E) on RSC-eval (left) and SSC-eval (right). For E2E we use confidence scores obtained with the "log-proba sum" configuration.

Figure 4 presents the Word Error Rate (WER) metric as a function of the percentage of filtered out data, for the two evaluated confidence scoring methods on RSC-eval and SSC-eval datasets. In the figure, both curves are obtained by sweeping the threshold across the entire range (0 to 1). Ideally we want to be in the lower left corner of the graphs, where we can select a whole corpus (0% filtered out) with 0% Word Error Rate. This is the hypothetical case, where the ASR system has no errors and correctly transcribed words are marked with a confidence score of 1.

As we are never in an ideal situation and ASR systems do have transcription errors, the curves show us how much we can select from newly acquired transcriptions, in percentage, based on the desired WER. The end-to-end system clearly offers an advantage on the SSC-dataset (orange curve). For example, if we can afford to use a new dataset with a WER of 4%, the E2E method enables us to select 80% of the dataset (20% is filtered out), while the MBR method allows us to select only 75% of the dataset. For the RSC dataset the two confidence estimation methods perform similarly for WERs between 1% and 2.5%, while the MBR method outperforms the E2E method if the user cannot afford WERs higher than 1%.

6. Conclusions

ASR systems are continuously migrating from the pipeline, lattice-based paradigm towards the end-to-end paradigm. At this point end-to-end ASR is performing similarly with classical ASR and the trend is clearly showing that it will outperform it soon.

Word-level confidence estimation for automatic speech recognition (ASR) is of crucial importance especially for commercial applications. However, at this moment word-level confidence estimation for end-to-end ASR was only addressed by two research works [10, 15].

In this paper we compared the state-of-the-art word-level confidence estimation method for lattice-based ASR with the newly proposed method for lattice-free ASR [10]. We showed that the two methods obtain similar results, with the first performing slightly better on read speech and the second performing slightly better on spontaneous speech. The experiments were carried out on several datasets in Romanian language.

In the near future we plan to perform this evaluation on standard English datasets and to extend the proposed confidence estimation method.

Acknowledgements

This work has been funded in part by the Operational Programme Human Capital of the Ministry of European Funds through the Financial Agreement 51675/09.07.2019, SMIS code125125, in part by a grant of the Romanian Ministry of Research and Innovation, CCCDI UEFISCDI, project number PN-III-P1-1.2-PCCDI-2017-0818 / 73PCCDI, within PNCDI III and in part by a grant of the Romanian Ministry of Education and Research, CNCS - UEFISCDI, project number PN-III-P1-1.1-PD-2019-0918, within PNCDI III.

R E F E R E N C E S

[1] *Szymański, Żelasko, Morzy, Szymczak, Żyła-Hoppe, Banaszczak, Augustyniak, Mizgajski and Carmie*l, "WER we are and WER we think we are", in Findings of the Association for Computational Linguistics: EMNLP 2020. 2020.

[2] *Jiang Hui, "*Confidence measures for speech recognition: A survey", in Speech communication, 45(4):455–470, 2005.

[3] *George Saon, Michael Picheny*, "Lattice-based Viterbi decoding techniques for speech translation.", In IEEE Automatic Speech Recognition and Understanding Workshop - ASRU, 2007.

[4] *Evermann, Gunnar, and Philip C. Woodland*. "Large vocabulary decoding and confidence estimation using word posterior probabilities." in IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings, 2000

[5] *Wessel, Frank, et al.* "Confidence measures for large vocabulary continuous speech recognition." IEEE Transactions on speech and audio processing (2001): 288-298, 2001.

[6] *Hossein Hadian, Hossein Sameti, Daniel Povey, and Sanjeev Khudanpur*, "End-to-end speech recognition using lattice-free MMI," in Interspeech, 2018, pp. 12–16.

[7] *Christoph Luscher, Eugen Beck, Kazuki Irie, Markus Kitza, Wilfried Michel, Albert Zeyer, Ralf Schluter, and Hermann Ney*, "RWTH ASR systems for LibriSpeech: Hybrid vs attention," in Interspeech 2019, pp. 231–235, 2019.

[8] *Zoltan T´uske, Kartik Audhkhasi, and George Saon*, "Advancing sequence-to-sequence based speech recognition," in Interspeech, 2019, pp. 3780–3784.

[9] *Shigeki Karita, Nanxin Chen, Tomoki Hayashi, Takaaki Hori, Hirofumi Inaguma, Ziyan Jiang, Masao Someki, Nelson Enrique Yalta Soplin, Ryuichi Yamamoto, Xiaofei Wang, Shinji Watanabe, Takenori Yoshimura, and Wangyou Zhang*, "A comparative study on transformer vs RNN in speech applications," in ASRU, 2019, pp. 449–456, 2019.

[10] *Dan Oneata, Alexandru Caranica, Adriana Stan, Horia Cucu,* "An Evaluation of Word-level Confidence Estimation for end-to-end Automatic Speech Recognition", In Proceedings of the 8th IEEE Spoken Language Technology Workshop (SLT 2021), Shenzhen, China, 2021.

[11] *Alexandru-Lucian Georgescu, Horia Cucu, Corneliu Burileanu,* "Kaldi-based DNN architectures *for* speech recognition in Romanian," in the Proceedings of the 10th Conference on Speech Technology and Human-Computer Dialogue (SpeD), 2019, Timișoara, Romania, 2019.

[12] *Andrey Malinin and Mark Gales,* "Uncertainty in structured prediction," arXiv preprint arXiv:2002.07650, 2020.

[13] *Woodward, A., Bonnín, C., Masuda, I., Varas, D., Bou-Balust, E., Riveiro, J.C.* "Confidence Measures in Encoder-Decoder Models for Speech Recognition" in Proc. Interspeech 2020, 611-615, DOI: 10.21437/Interspeech, 2020.

[14] *Apoorv Vyas, Pranay Dighe, Sibo Tong, and Herve Bourlard,* "Analyzing uncertainties in speech recognition using dropout," in ICASSP, 2019, pp. 6730–6734

[15] *David Qiu, Qiujia Li, Yanzhang He, Yu Zhang, Bo Li, Liangliang Cao, Rohit Prabhavalkar, Deepti Bhatia, Wei Li, Ke Hu, Tara N. Sainath, Ian McGraw,* "Learning Word-Level Confidence For Subword End-to-End ASR", ICASSP 2021, arXiv:2103.06716, 2021.

[16] *Kaldi Speech Recognition Toolkit*: https://github.com/kaldi-asr/kaldi, accessed January 2021.

[17] *ESPnet: end-to-end speech processing toolkit*: https://github.com/espnet/espnet, accessed January 2021.

[18] *Dan Hendrycks and Kevin Gimpel,* "A baseline for detecting misclassified and out-of-distribution examples in neural networks," in ICLR, 2016.

[19] *Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov*, "Dropout: A simple way to prevent neural networks from overfitting," Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929–1958, 1 2014.

[20] *P. J. Bickel and K. A. Doksum,"*Mathematical Statistics: Basic Ideas and Selected topics.", HoldenDay Inc., Oakland, CA, USA, 1977.

[21] *Alexandru-Lucian Georgescu, Horia Cucu, Andi Buzo, Corneliu Burileanu,* "RSC: A Romanian Read Speech Corpus for Automatic Speech Recognition," in the Proceedings of The 12th Language Resources and Evaluation Conference (LREC), pp. 6606-6612, 2020, Marseille, France.

[22] *A-L. Georgescu, C. Manolache, D. Oneaţă, H. Cucu and C. Burileanu,* "Data-Filtering Methods for Self-Training of Automatic Speech Recognition Systems," 2021 IEEE Spoken Language Technology Workshop (SLT), Shenzhen, China, 2021, pp. 1-7, doi: 10.1109/SLT48900.2021.9383577, 2021.