INVESTIGATING ROBUSTNESS OF A DEEP ASR PERFORMANCE PREDICTION SYSTEM

Zied Elloumi, Olivier Galibert, Benjamin Lecouteux, Laurent Besacier

To cite this version:

Zied Elloumi, Olivier Galibert, Benjamin Lecouteux, Laurent Besacier. INVESTIGATING ROBUSTNESS OF A DEEP ASR PERFORMANCE PREDICTION SYSTEM. [Research Report] LIG lab; LNE. 2019. hal-02131931

HAL Id: hal-02131931
https://hal.archives-ouvertes.fr/hal-02131931

Submitted on 16 May 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
INVESTIGATING ROBUSTNESS OF A DEEP ASR PERFORMANCE PREDICTION SYSTEM

Zied Elloumi 1, 2, Olivier Galibert 1, Benjamin Lecouteux 2, Laurent Besacier 2

1 Laboratoire national de métrologie et d’essais (LNE), France
2 Univ. Grenoble Alpes, CNRS, Grenoble INP, LIG, F-38000 Grenoble, France
zied.elloumi@univ-grenoble-alpes.fr, laurent.besacier@univ-grenoble-alpes.fr

ABSTRACT

In this paper, we address a relatively new task: prediction of ASR performance on unseen broadcast programs with ASR system considered like a black-box. In a previous study, we compared two different prediction approaches: a baseline performance prediction based on engineered features and a new strategy based on learnt features using CNNs which combines both textual (ASR-transcription) and signal inputs. In this new contribution, we analyze more deeply the robustness of both ASR prediction approaches (learnt and engineered features) by studying the effect of speech style, training set size and ASR system considered a training or test time. Performance prediction is shown to be more difficult on spontaneous speech. Effect of training size of the predictor is also investigated and it is found that while CNN predictor is better than the baseline predictor, it is also more sensible to training size reduction. Finally, we investigate the robustness of error prediction when the predictor is trained with outputs of a particular ASR system and used to predict performance on unseen broadcast programs and unseen (new) ASR system.

Index Terms— ASR Performance Prediction, Large Vocabulary Continuous Speech Recognition, TV shows, Convolutional Neural Networks

1. INTRODUCTION

Predicting automatic speech recognition (ASR) performance on unseen speech recordings is an important Grail of speech research. From a research point of view, such a task helps understanding automatic (but also human) transcription performance variation and its conditioning factors. From a technical point of view, predicting ASR difficulty is useful in application workflows where transcription systems have to be quickly built (or adapted) to new document types (predicting learning curves, estimating amount of adaptation data needed to reach an acceptable performance, etc.).

Related works Other works propose to use more features types than acoustic, [2] exploit ASR, textual, hybrid and acoustic features to predict a WER on different conditions. By exploiting previous works in ASR and machine translation performance prediction tasks [2, 3, 4, 5], [6] proposed an open-source tool named TranscRater based on feature extraction (lexical, syntactic, signal and language model features) and regression (WER prediction) or classification (if multiple ASR outputs are provided). Evaluation was performed on CHiME-3 data. For both regression and classification tasks, it was shown that signal features did not help WER prediction. Finally, [1] proposed a new ASR performance prediction approach based on CNN. It is based on both textual and raw signal features. Evaluation was performed on a French corpus of TV programs. We give more details on this work and analyze our results more deeply in the next sections.

Contribution Extending our previous work on ASR-performance prediction (PP) task [1], the current work investigates the robustness of PP systems evaluated on unseen broadcast programs. Firstly, we present a large and heterogeneous French corpus (containing non spontaneous and spontaneous speech), an evaluation framework, as well as both engineered features and learnt features approaches dedicated to performance prediction task. In this study, we focus only on the combination of both textual (ASR transcription) and speech signal, while, ASR system is considered as a black-box. Secondly, we propose a deep analysis in order to evaluate the robustness of PP systems by studying: i) the effect of speech style on predictor system quality, ii) the influence of training set (for PP) size on ASR performance prediction systems, iii) the robustness of error prediction when the predictor is trained with outputs of a particular ASR system and used to predict performance on shows transcribed with a different ASR system.

Outline The paper is organized as follows. Section 2 details our evaluation framework. Section 3 presents both ASR performance prediction approaches. Section 4 is a deep analysis of the robustness of PP approaches by studying the effect of speech style, training set size and ASR system considered. Finally, section 5 concludes this work.

2. FRAMEWORK FOR ASR-PERFORMANCE PREDICTION

We focus on ASR performance prediction on unseen speech data. Our hypothesis is that performance prediction systems should only use ASR transcripts (and the signal) as input
in order to predict the corresponding transcription quality (WER). Obviously, reference (human) transcriptions are only available at training of the prediction system. A Train\textsubscript{pred} corpus contains many pairs \{ASR output, Performance\} (more than 75k ASR turns in this work), a Test\textsubscript{pred} corpus only contains ASR outputs (more than 6.8k turns in this work) and we try to predict the associated transcription performance. Reference (human) transcriptions on Test\textsubscript{pred} are used to evaluate prediction quality. In order to evaluate WER prediction task, we use Mean Absolute Error (MAE) metric.

Data

The data used in our protocol comes from different broadcast collections in French: Quaero\textsuperscript{1}, ETAPE [7], ESTER 1 & ESTER 2 [8] and REPERE [9]. As described in Table 1, the full data contains non spontaneous speech (NS) and spontaneous speech (S). The data used to train our ASR system (Train\textsubscript{Acoustic}) is selected from the non spontaneous speech style that corresponds mainly to broadcast news. The Train\textsubscript{Acoustic} data set were unseen in the Train\textsubscript{Pred}. Moreover, more challenging (high WERs) shows were selected for Test\textsubscript{Pred}.

### Table 1: Distribution of our data set between non-spontaneous (NS) and spontaneous (S) styles

<table>
<thead>
<tr>
<th>Acoustic</th>
<th>Train\textsubscript{Acoustic}</th>
<th>Train\textsubscript{Pred}</th>
<th>Test\textsubscript{Pred}</th>
</tr>
</thead>
<tbody>
<tr>
<td>NS</td>
<td>100h51</td>
<td>30h27</td>
<td>04h17</td>
</tr>
<tr>
<td>S</td>
<td>74h27</td>
<td>59h25</td>
<td>04h42</td>
</tr>
<tr>
<td>Duration</td>
<td>100h51</td>
<td>89h52</td>
<td>108h25</td>
</tr>
</tbody>
</table>

ASR systems

To obtain speech transcripts (ASR outputs) for the prediction model with different qualities, we built our own French ASR systems based on the KALDI toolkit [10]. For the acoustic modelling (AM), we used Train\textsubscript{Acoustic} dataset (100 hours of broadcast news from ESTER, REPERE, ETAPE and Quaero) to learn 3 acoustic models (following a standard Kaldi recipe) with 13 dimensions mel-frequency cepstral coefficients (MFCC). These acoustic models are named and trained as following: i) GMM: we learnt triphone models with GMM distributions; ii) SGMM: we learnt triphone models with SGMM (subspace gaussian mixture models) distributions; iii) DNN: we learnt a hybrid HMM/DNN system using DNNs of 4 hidden layers (with 1024 units).

For language modelling (LM), we use both 3-gram and 5-gram language models trained on several French corpora\textsuperscript{2} using SRILM toolkit [11]. For the pronunciation model, we used lexical resource BDLEX [12] as well as automatic grapheme-to-phoneme (G2P)\textsuperscript{3} transcription to find pronunciation variants of our vocabulary (limited to 80k). Finally, the LNE-Tools [13] are used to evaluate the ASR performance in terms of Word Error Rate (WER), knowing that overlapped speech and empty utterances are removed.

### Table 2: Description of 4 ASR systems produced and their WER performance evaluated on our Train\textsubscript{Pred} and Test\textsubscript{Pred} sets

<table>
<thead>
<tr>
<th>ASR systems</th>
<th>AM</th>
<th>LM</th>
<th>Train\textsubscript{Pred} WER (%)</th>
<th>Test\textsubscript{Pred} WER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASR1 [1]</td>
<td>DNN</td>
<td>5-gram</td>
<td>22.29</td>
<td>31.20</td>
</tr>
<tr>
<td>ASR2</td>
<td>DNN</td>
<td>3-gram</td>
<td>23.64</td>
<td>32.80</td>
</tr>
<tr>
<td>ASR3</td>
<td>SGMM</td>
<td>3-gram</td>
<td>24.58</td>
<td>34.01</td>
</tr>
<tr>
<td>ASR4</td>
<td>GMM</td>
<td>3-gram</td>
<td>27.02</td>
<td>36.79</td>
</tr>
</tbody>
</table>

3. ASR-PERFORMANCE PREDICTION SYSTEMS

3.1. Engineered features based

An open-source tool for automatic speech recognition quality estimation, TranscRater [6], is used for the baseline regression approach (named as TR system in our experiments). It exploits Extremely Randomized Trees algorithm [14] which is a very competitive algorithm in WER prediction and successfully used in [2, 3, 4, 5]. Features selection was performed using Randomized Lasso [15]. TranscRater requires engineered features to predict the WER performance. These features are extracted for each utterance and are of several types: Part-of-speech (POS) features capture the plausibility of the transcription from a syntactic point of view;\textsuperscript{4} Language model (LM) features capture the plausibility of the transcription according to a N-gram model (fluency);\textsuperscript{5} Lexicon-based (LEX) features are extracted from the ASR lexicon;\textsuperscript{6} Signal (SIG) features capture the difficulty of transcribing the input signal (general recording conditions, speaker-specific accents).\textsuperscript{7} This approach, based on engineered features, One

---

\textsuperscript{1}http://www.quaero.org

\textsuperscript{2}3323M words in total - from EBuchshop, TED2013, Wiit, GlobalVoices, Gigaword, Europarl-v7, MultiUN, OpenSubtitles2016, DGT, News Commentary, News WMT, LeMonde, Trames, Wikipedia and transcriptions of our Train\textsubscript{Acoustic} dataset

\textsuperscript{3}https://goo.gl/NCwpxz

\textsuperscript{4}Treetagger [16] is used for POS extraction in this study

\textsuperscript{5}We train a 5-gram LM on 3323M words text already mentioned

\textsuperscript{6}A feature vector containing the frequency of phoneme categories in its pronunciation is defined for each input word

\textsuperscript{7}For feature extraction, TranscRater computes 13 MFCC, their delta, acceleration and log-energy, F0, voicing probability, loudness contours and pitch for each frame. The SIG feature vector for the entire input signal is obtained by averaging the values of each frame
drawback is that its application to new languages requires adequate resources, dictionaries and tools which makes the prediction method less flexible.

3.2. Learnt features based

In [1], we proposed a new approach using convolution neural networks (CNNs) to predict ASR performance from a collection of heterogeneous broadcast programs (both radio and TV). We particularly focused on the combination of text (ASR transcription) and signal (raw speech) inputs which both proved useful for CNN prediction. We also observed that our system remarkably predicts WER distribution on a collection of speech recordings. The network input can be either a pure text input, a pure signal input (raw signal) or a dual (text+speech) input. To avoid memory issues, signals are downsampled to 8khz and models are trained on six-second speech turns (shorter speech turns are padded with zeros). For text input, the architecture is inspired from [17]: the input is a matrix of dimensions 296x100 (296 is the longest ASR hypothesis length in our corpus; 100 is the dimension of pre-trained word embeddings on a large held out text corpus of 3.3M words). For speech input, we use the best architecture (m18) proposed in [18] of dimensions 48000 x 1 (48000 samples correspond to 6s of speech). For text input, the architecture is inspired from [17]: the input is a matrix of dimensions 296x100 (296 is the longest ASR hypothesis length in our corpus; 100 is the dimension of pre-trained word embeddings on a large held out text corpus of 3.3M words). For speech input, we use the best architecture (m18) proposed in [18] of dimensions 48000 x 1 (48000 samples correspond to 6s of speech). For text input, the architecture is inspired from [17]: the input is a matrix of dimensions 296x100 (296 is the longest ASR hypothesis length in our corpus; 100 is the dimension of pre-trained word embeddings on a large held out text corpus of 3.3M words). For speech input, we use the best architecture (m18) proposed in [18] of dimensions 48000 x 1 (48000 samples correspond to 6s of speech). For text input, the architecture is inspired from [17]: the input is a matrix of dimensions 296x100 (296 is the longest ASR hypothesis length in our corpus; 100 is the dimension of pre-trained word embeddings on a large held out text corpus of 3.3M words). For speech input, we use the best architecture (m18) proposed in [18] of dimensions 48000 x 1 (48000 samples correspond to 6s of speech). For text input, the architecture is inspired from [17]: the input is a matrix of dimensions 296x100 (296 is the longest ASR hypothesis length in our corpus; 100 is the dimension of pre-trained word embeddings on a large held out text corpus of 3.3M words). For speech input, we use the best architecture (m18) proposed in [18] of dimensions 48000 x 1 (48000 samples correspond to 6s of speech).

4. DEEP ANALYSIS OF OUR PROPOSED APPROACH

4.1. Effect of speech style on ASR performance prediction quality

In order to better understand the behavior of the systems for different conditioning factors, we propose in this section to analyze the effect of speech style on PP outputs at broadcast show instance level and at speech style level.

In Figure 1, we compare TR and CNN systems in terms of MAE (CNN is better when $\Delta_{M^AE} > 0$) on Test$_1$ (ASR1) dataset at broadcast show instance level and for both NS (green) and S (red) speech styles.

82/102 broadcast show instances by a large margin (50 show instances present a $\Delta_{M^AE}$ larger than 5%).

Fig. 1: Evaluation of TR and CNN systems in terms of $\Delta_{M^AE}$ (CNN is better when $\Delta_{M^AE} > 0$) on Test$_1$ (ASR1) dataset at broadcast show instance level and for both NS (green) and S (red) speech styles.

In Figure 2, we compare both CNN and TR systems in terms of MAE on Test$_1$ (ASR1) set at broadcast program level. The performance obtained show that Spontaneous (S) is more difficult to predict the performance than Non Spontaneous (NS) speech style. In Spontaneous part, we notice that the gap between CNN and TR curve is wider than for Non Spontaneous speech. That means that CNN is able to predict a high WER, while TR predicts a performance around the mean WER observed on training data [1]. To confirm this hypothesis, we created an artificial reference by attributing the mean WER observed on training data (22.29%) to all utterances. Evaluating our systems’ outputs with this basic reference lead to the following MAE scores: 13.15% and 21.58% on TR and CNN systems respectively, which confirms our intuition.

Fig. 2: Evaluation of PP system on Test$_1$ (ASR1) dataset in terms of MAE at broadcast program level.

4.2. Effect of training set size on the quality of ASR performance prediction

Training-set size and its influence on systems’ quality remains always an important issue for many tasks (speech recognition, machine translation, image classification, etc). In this section, we attempt to understand what is the effect of training set size on our PP systems (TR and CNN). We build new ASR performance prediction systems with less training data using subsets of Train$_1$ (ASR1). We selected randomly 20% (overall WER of 21.50%) and 50% (overall WER of 22.40%) of the full Train$_1$ (ASR1). PP systems using engineered features
We emphasize on the fact that all evaluation sets (Test\textsubscript{i}) order to measure robustness of PP systems in terms of MAE on CNN systems) learnt on 100%, 50% and 20% of the whole datasets Test\textsubscript{i} (ASR systems i = 1, 2, 3, 4) and apply them to Test\textsubscript{i} sets. We obtain a 4x4 matrix of results for each PP system. Results are given in Table 5 and Table 6.

### 4.3. Effect of ASR output quality at training time for performance prediction

In previous sections, we used ASR\textsubscript{1} system to obtain speech transcripts and learn PP systems. In this section, we aim to investigate the effect of ASR output quality at training time for performance prediction. We learn 4 PP systems for each prediction approach named TR\textsubscript{i} and CNN\textsubscript{i} using speech transcripts of Train\textsubscript{i}, (ASR systems i = 1, 2, 3, 4) and apply them to Test\textsubscript{i} sets. We obtain a 4x4 matrix of results for each PP system. Results are given in Table 5 and Table 6.

If we focus on the difference between evaluation sets (lines), results show that Test\textsubscript{1} obtained the best prediction in terms of MAE on CNN and TR systems, knowing that Test\textsubscript{1} (average WER of 31.20%) has the best ASR output quality in table 2. We also notice that ASR output quality (see Table 2) and PP system quality seem correlated (when ASR quality is lower - eg i = 4 - MAE of PP systems increases). This confirms the trend, already noticed for spontaneous speech, that it is harder to predict higher WERs. Anyway, it is interesting to note that a PP system learnt for a particular ASR system (ASR\textsubscript{1} for instance) is not too much degraded when applied on ASR outputs obtained with a different transcription system (ASR\textsubscript{i} for i = 2, 3, 4 for instance).

Looking at the amount of training data factor (columns), we observe that reducing training set size increases MAE for the CNN system. For example, on Test\textsubscript{1} set, we obtained respectively 19.24% and 21.53% on CNN-100% and CNN-20% systems in terms of MAE. It means that training set size have a strong influence on the performance of the PP system based on CNNs. Unlike CNNs, Table 3 shows that TR approach is not too much degraded when training size decreases (surprisingly TR-20% has better quality than TR-100% !).

### 5. CONCLUSION

The main result of this experiment is that both PP systems (CNN and TR) are rather stable whatever the ASR output quality is at training time. It is remarkable to note that CNN\textsubscript{4} system trained on Train\textsubscript{4} is actually slightly better to predict performance on unseen broadcast programs transcribed with better ASR systems: the last line of Table 6 displays better MAE on Test\textsubscript{2}, Test\textsubscript{3} and Test\textsubscript{4}. This result (robustness of PP systems to ASR quality at both training and test time) is important for the portability and application of performance prediction systems in practical scenarios.

### Table 4: Evaluation of new CNN systems on 4 evaluation datasets Test\textsubscript{i} (ASR\textsubscript{i}) in terms of MAE

<table>
<thead>
<tr>
<th>Evaluation sets</th>
<th>CNN-100%</th>
<th>CNN-50%</th>
<th>CNN-20%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test\textsubscript{1}</td>
<td>19.24</td>
<td>20.55</td>
<td>21.53</td>
</tr>
<tr>
<td>Test\textsubscript{2}</td>
<td>19.67</td>
<td>20.79</td>
<td>21.87</td>
</tr>
<tr>
<td>Test\textsubscript{3}</td>
<td>20.64</td>
<td>21.70</td>
<td>22.90</td>
</tr>
<tr>
<td>Test\textsubscript{4}</td>
<td>21.34</td>
<td>22.44</td>
<td>23.62</td>
</tr>
</tbody>
</table>

### Table 5: Effect of ASR output quality at training time for performance prediction - TR systems evaluated with MAE

<table>
<thead>
<tr>
<th>PP systems</th>
<th>Test\textsubscript{1}</th>
<th>Test\textsubscript{2}</th>
<th>Test\textsubscript{3}</th>
<th>Test\textsubscript{4}</th>
</tr>
</thead>
<tbody>
<tr>
<td>TR\textsubscript{1}</td>
<td>21.99</td>
<td>22.15</td>
<td>23.33</td>
<td>23.00</td>
</tr>
<tr>
<td>TR\textsubscript{2}</td>
<td>21.68</td>
<td>21.72</td>
<td>22.67</td>
<td>22.33</td>
</tr>
<tr>
<td>TR\textsubscript{3}</td>
<td>21.62</td>
<td>21.67</td>
<td>22.37</td>
<td>22.13</td>
</tr>
<tr>
<td>TR\textsubscript{4}</td>
<td>21.58</td>
<td>21.60</td>
<td>22.66</td>
<td>21.95</td>
</tr>
</tbody>
</table>

### Table 6: Effect of ASR output quality at training time for performance prediction - CNN systems evaluated with MAE

<table>
<thead>
<tr>
<th>PP systems</th>
<th>Test\textsubscript{1}</th>
<th>Test\textsubscript{2}</th>
<th>Test\textsubscript{3}</th>
<th>Test\textsubscript{4}</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN\textsubscript{1}</td>
<td>19.24</td>
<td>19.67</td>
<td>20.64</td>
<td>21.34</td>
</tr>
<tr>
<td>CNN\textsubscript{2}</td>
<td>19.75</td>
<td>19.78</td>
<td>20.54</td>
<td>21.18</td>
</tr>
<tr>
<td>CNN\textsubscript{3}</td>
<td>19.87</td>
<td>19.81</td>
<td>20.62</td>
<td>21.39</td>
</tr>
<tr>
<td>CNN\textsubscript{4}</td>
<td>19.26</td>
<td>19.28</td>
<td>19.94</td>
<td>20.22</td>
</tr>
</tbody>
</table>
6. REFERENCES


