Increasing Context for Estimating Confidence Scores in Automatic Speech Recognition

Anton Ragni, *Member, IEEE,* Mark Gales, *Fellow, IEEE,* Oliver Rose, Kate Knill, *Member, IEEE,* Alexandros Kastanos, Qiujia Li, *Member, IEEE,* and Preben Ness

***Abstract*—Accurate confidence measures for predictions from machine learning techniques play a critical role in the deployment and training of many speech and language processing applications. For example, confidence scores are important when making use of automatically generated transcriptions in training automatic speech recognition (ASR) systems, as well as down-stream applications, such as information retrieval and conversational assistants. Previous work on improving confidence scores for these systems has focused on two main directions: designing features correlated with improved confidence prediction; and employing sequence models to account for the importance of contextual information. Few studies, however, have explored incorporating contextual information more broadly, such as from the future, in addition to the past, or making use of alternative multiple hypotheses in addition to the most likely one. This article introduces two general approaches for encapsulating contextual information from lattices. Experimental results illustrating the importance of increasing contextual information for estimating confidence scores are presented on a range of limited resource languages where word error rates range between 30% and 60%. The results show that the novel approaches provide significant gains in the accuracy of confidence estimation.**

***Index Terms*—Speech recognition, confidence, recurrent neural network, attention, graph structures.xf**

I. Introduction

AUTOMATIC speech recognition (ASR) accuracy has seen a gradual but consistent improvement across a wide range of domains in recent years. The use of transcriptions as is, however, typically leads to a poor performance in applications utilising ASR technology (downstream tasks) as mistakes cannot be flagged and acted upon. If a measure indicating the likelihood of a mistake occurring can be provided along with each transcribed word[1], then ASR technology could be applied to a wider range of domains where near-perfect transcriptions are not possible. Such measures are also very useful for the development of ASR technology itself (upstream tasks). For instance, semi-supervised training [3], speaker adaptation [4], system combination [5] and the transcription process itself [6] can all benefit from the knowledge of transcription mistakes.

Confidence scores, which are often presented as a numeric value between 0 and 1 for each word, have been widely used in this role since the 1990s. A number of schemes have been proposed for estimating these confidence scores. These range from simple schemes based on word posterior probabilities [7]–[9] to more complex schemes that utilise powerful sequence models, such as conditional random fields [10] and recurrent neural networks (RNN) [2], [11]–[13]. Amongst them, the approaches based on word posterior probabilities have become the most commonly used, and successful, schemes. Despite significant research into finding more accurate alternatives, these word posterior probabilities have proven to be a very challenging baseline to improve upon.

This article demonstrates that *context* plays a critical role in accurate confidence estimation. It introduces two novel approaches that provide two different systematic ways for incorporating information from more general than sequences graph-like, lattice, data structures. The *first approach* extends RNNs from sequences to lattices by means of an attention mechanism [14] that enables information from multiple paths to be combined and propagated, which is impossible with standard RNNs[2]. The *second approach* leverages the flexibility offered by attention to combine information more generally, such as from all overlapping in time path segments, an entire lattice, or a set of lattices. Both approaches lead to a significant increase in the amount of contextual information available for confidence predictions and yield significant improvements over word posterior probabilities, as illustrated by an extensive evaluation in challenging limited resource conditions.

The rest of this article is organised as follows. Section II relates the proposed approaches to other work in the audio, speech and language processing area. The following Section III provides an overview of conventional confidence score estimation approaches. Section IV introduces the proposed recurrent and attention methods for extracting contextual information for confidence estimation from lattices. Experiments with the proposed methods are presented in Section V. Finally, conclusions drawn from this work are given in Section VI.

All authors were with the Department of Engineering, University of Cambridge, Cambridge, CB2 1PZ, UK. A. Ragni is now with the Department of Computer Science, University of Sheffield, Sheffield, S10 2TN, UK, e-mail: (see https://www.sheffield.ac.uk/dcs/people/academic/anton-ragni).

All authors were supported in part by the ALTA Institute, Cambridge University. A. Ragni and M. Gales were also supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via Air Force Research Laboratory (AFRL) contract # FA8650-17-C-9117. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, AFRL or the U.S. Government. The U.S. Government is authorised to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

[1]For simplicity, this article excludes handling deletion errors. Approaches for taking deletion errors into account have been proposed [1] and can be incorporated into all of the models described in this article following [2].

[2]A short summary of this approach has been previously presented in [15].

II. Relation to prior work

The importance of context in confidence estimation has been appreciated for a long time. Even the conventional methods discussed in Section III make use of the whole hypothesised transcript to estimate confidence. Furthermore, the accuracy of some of those methods have been long linked with the number of alternative transcripts used in their estimation [7]. Alternative transcripts have so far been exploited only for extracting new types of features. These range from a hypothesis density [16], [17], which represents the quantity and/or diversity of alternative hypotheses, to learnt fixed or variable length lattice embeddings [18]–[21]. The recurrent and attention methods proposed in this work differ from this line of research by learning embeddings and confidence scores of individual arcs in a single fully integrated framework.

The context itself can be viewed more broadly than just alternative transcripts generated by a single ASR system. It is common for high-performing ASR systems to combine multiple diverse sub-systems using approaches such as ROVER [5] and confusion network (CN) combination (CNC) [22] to reduce transcription error. The proposed attention method applied to multiple hypotheses or CNs can be viewed as a more general trainable solution. Furthermore, unlike the dynamic programming algorithms used by ROVER and CNC, the attention method can be generalised to more general graph structures, such as lattices, as will be illustrated in Section V.

Alternative ASR systems and features such as hypothesis density provide important information about how consistent or stable any prediction is. The notion of stability gave rise to alternative language model assessment criteria [23], data augmentation methodologies [24] as well as confidence estimation approaches [16]. The proposed attention method can be viewed as a more general form of acoustic stability [16], where lattices rather than one-best hypotheses are used and a trainable attention mechanism replaces counting how many alternative words emerged by perturbing acoustic and/or language model scales. This novel form of acoustic stability offers a range of advantages. In particular, the attention mechanism provides for a more nuanced definition of stability that can take into account temporal (e.g. overlap), topological (e.g. location and connectivity), and semantic (e.g. word) information.

The generality of graph-like data structures makes them a popular choice of data representation in many other areas of audio, speech and language processing. Recurrent neural networks with such a complex input have been previously examined by a number of authors [19], [25]–[27]. The key difference between those extensions and the proposed recurrent method is the use of attention for computing history states, rather than averaging, pooling or gating. A broader application of attention to graph-like data structures has been explored in [28], [29], where various generalisations of adjacency (connectivity) matrices were examined for encoding topologies of machine translation (MT) lattices into a matrix format. The proposed attention method extends that line of work to ASR lattices, which contain information not present in MT lattices such as time. The new information enables novel forms of adjacency matrices and attention to be explored as will be described in Section IV.

III. Conventional methods

This section will focus on hidden Markov model (HMM) based ASR systems as the problem of miscalibrated predictions of multi-class (softmax) classifiers employed by alternative end-to-end (E2E) approaches [30]–[32] has a long history in machine learning and have been extensively covered elsewhere [33], [34]. Furthermore, some of the approaches discussed here can be used with E2E systems.

A. Word posterior probabilities

Word posterior probabilities emerged as a popular approach for obtaining confidence scores at the end of the 1990s [16]. One of the key reasons for their popularity is the simplicity of computing word posterior probabilities from lattices generated by HMM-based recognisers. A lattice, illustrated in Figure 1, is a popular encoding format used in HMM-based ASR to retain a vast number of hypothesised transcriptions. Similar

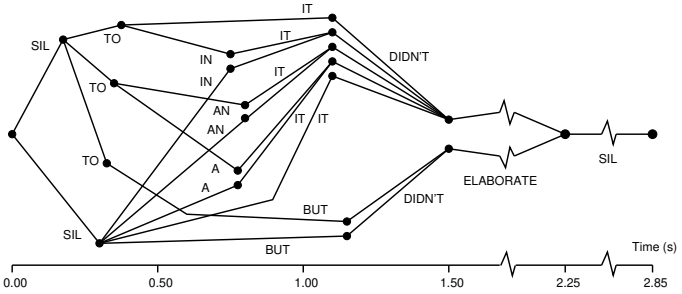

Fig. 1: An example of a lattice (HMM case)

structures have been explored for E2E speech recognition [35], [36]. Given a lattice, a forward-backward algorithm [37] can be applied to estimate posterior probabilities of lattice arcs, or edges [9], [16]. The forward and backward probability associated with lattice arc $\boldsymbol{e}_i$ can be computed recursively by

$$\alpha_i = \sum_{j \in \overrightarrow{\mathcal{N}}_i^{(1)}} \alpha_j s_i \quad \text{and} \quad \beta_i = \sum_{j \in \overleftarrow{\mathcal{N}}_i^{(1)}} \beta_j s_j \tag{1}$$

where $\overrightarrow{\mathcal{N}}_i^{(1)}$ and $\overleftarrow{\mathcal{N}}_i^{(1)}$ is the set of arcs which are direct left and right neighbours of $\boldsymbol{e}_i$ respectively and s_i is an arc score. Given a pair of forward α_i and backward β_i probabilities, the arc posterior probability p_i can be computed by

$$p_i = \frac{1}{[[\mathcal{L}]]} \alpha_i \beta_i = P(\boldsymbol{e}_i | \boldsymbol{O}) \tag{2}$$

where $[[\mathcal{L}]]$ is a lattice weight (forward probability of the final arc or backward probability of the initial arc[3]). Lattice arc posterior probabilities are not the same as word posterior probabilities [9]. A number of different schemes have been proposed for deciding how to optimally combine and normalise arc posterior probabilities to yield accurate estimates of word posterior probabilities [7]–[9]. In particular,

[3]Multiple initial/final arcs can be handled either by summing over their probabilities or adding new preceding/following arcs to ensure that only one initial/final arc exists.

the confusion network (CN) approach [7], [8] clusters lattice arcs first based on time and then based on their word labels. The time clustering creates a chain like structure, where consecutive nodes, or bins, $\mathcal{C}_{i-1}$ and $\mathcal{C}_i$ are connected by one or more lattice arcs. The word clustering then merges any arcs with identical word labels. Figure 2 shows an example of a confusion network produced from the lattice in Figure 1. The

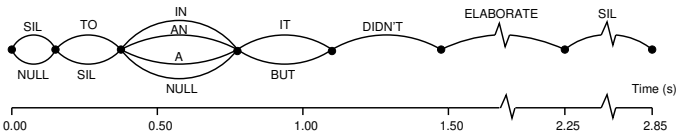

Fig. 2: An example of a confusion network (HMM case)

probability of word w in bin $\mathcal{C}_i$ is computed by

$$p_{i,w} = \frac{\sum_{j \in \mathcal{C}_i} p_j \delta(w, w_j)}{\sum_{j \in \mathcal{C}_i} p_j} = P(w|\mathcal{C}_i, \boldsymbol{O}) \tag{3}$$

where $\delta(w, w_j) = 1$ if $w = w_j$ and 0 otherwise. These probabilities are used as estimates of word posterior probabilities.

Word posterior probabilities derived from lattices are often criticised for providing overly optimistic estimates of confidence. This problem exists with both Gaussian mixture model [7] as well as neural network based HMMs (e.g. [2]). One of the major contributing issues is the limited number of arcs generated by speech recognisers, which leads to smaller than expected denominator terms in equation (2). Another issue is the underlying statistical models themselves [33], [38]. The problem of poor confidence estimates in neural network based classifiers is the subject of active research [33], [39]. Approaches for rectifying this problem can be divided into two groups. The first group comprises approaches that make changes to the model architecture [33], [40] and/or modify the standard parameter estimation methodology [41], [42]. Popular examples included temperature scaling [33], ensembles [39], and data augmentation [21]. The second group comprises *post-hoc* calibration approaches that transform predictions such that they exhibit a more favourable behaviour [34], [43], [44]. This last group of approaches is more general, as it supports both HMM and E2E ASR systems. Common approaches in this group also include (piece-wise) linear mappings [7], feed-forward [45] and more complex [46] neural network models.

B. Sequence models

One critical issue with the post-hoc calibration schemes mentioned in the previous section are strong independence assumptions, which disregard the sequential nature of speech and confidence estimation. Discriminative graphical models [47] emerged as a powerful alternative to HMMs for modelling posterior probabilities of word sequences given observation sequences [48]–[50]. Many such approaches are based on conditional random fields (CRF) [51] which enable computing word posterior probabilities using the efficient forward-backward algorithm. These probabilities then can be used as confidence scores [10]. Even though CRFs theoretically enable arbitrary long dependencies in observations to be modelled, it is not obvious how to extract and model them. The recent revival of interest in neural network approaches has led to exploring recurrent neural networks (RNN) for confidence prediction. The key element of an RNN is a recursively updated history state

$$\boldsymbol{h}_i = \boldsymbol{\phi}(\boldsymbol{h}_{i-1}, \boldsymbol{x}_i) \tag{4}$$

where $\boldsymbol{h}_{i-1}$ and $\boldsymbol{h}_i$ are the past and current history state, $\boldsymbol{x}_i$ are features associated with the current position in a sequence, $\boldsymbol{\phi}$ is a non-linear transform, such as [52], [53]. The recursive nature of history states, where any state depends on all past features, provides an opportunity for capturing long-range dependencies. It is also possible to extend this approach to capturing future dependencies using a bi-directional RNN [54], where an additional, future, state is employed. In either case, confidence scores can be predicted by learning a suitable non-linear transformation of RNN history states. Both uni- and bi-directional RNNs have been explored for predicting confidence scores [2], [11], [13], [55]. RNNs have also been exploited within energy-based models to yield sequence-level, or utterance, confidence scores [56].

The ability of CRFs and RNNs to yield accurate confidence scores also critically relies on the availability of informative features. The long history of confidence scores in ASR has led to the development of a large number of features that have been found useful for confidence estimation.

a) Acoustic features: Given a segment of speech, the simplest kind of features that can be extracted are duration [17], speaking rate and signal-to-noise ratio [57], the first and higher order statistics, HMM likelihoods [17], [57] and other dynamic kernels [50], [58], [59]. Powerful approaches from deep learning include various forms of encoders [14], [53], [60] that enable general mappings from variable length observation sequences to a fixed length to be learnt.

b) Language features: Similar approaches have been adopted with word features. These include count-based and n-gram order features [45], [57] and simple generative kernels, such as language model log-probabilities [17], [45]. Popular features from representation learning and the deep learning area include word embeddings [61].

c) Lexicon features: These features aim to extract information at a finer, subword, level. There are a number of possible subword units to consider, such as graphemes, phonemes, syllables, morphs and n-gram extensions of these units. Each unit can benefit from all of the features discussed above (both acoustic and language features, e.g. [62]).

d) Graph features: Features examined so far were focused on deriving information at a local segment/word level. Given a sequence or graph output representation, it is possible to extract features that reflect the global context. The word posterior probabilities discussed in this section are examples of graph features [45]. Other examples include arc/node density and stability of hypothesised word with respect to the acoustic or language model scale, or acoustic stability [16], [17].

C. Training and evaluation

Confidence prediction models are commonly trained by minimising (binary) cross-entropy (CE)

$$H(\boldsymbol{c}, \boldsymbol{c}^*) = -\frac{1}{T}\sum_{t=1}^{T} c_t^* \log(c_t) + (1 - c_t^*)\log(1 - c_t) \quad (5)$$

where $\boldsymbol{c}$ and $\boldsymbol{c}^*$ are predicted and reference confidence scores respectively. The reference confidence scores are not immediately available and must be inferred. Given a hypothesis $\boldsymbol{w}$ and reference $\boldsymbol{w}^{\mathtt{ref}}$ word sequence, the alignment between these sequences can be obtained using a Levenshtein algorithm [5]

$$\overline{L}_i^j = \min\left\{\overline{L}_{i-1}^{j-1} + L_{i-1,i}^{j-1,j}, \overline{L}_{i-1}^{j} + L_{i-1,i}, \overline{L}_{i}^{j-1} + L^{j-1,j}\right\} \quad (6)$$

where $\overline{L}_i^j$ is a cumulative loss incurred on reaching position i in the first sequence and position j in the second sequence, $L_{i-1,i}$ and $L^{j-1,j}$ are losses incurred on making a single step transition from one position to the next in either the first or the second sequence, $L_{i-1,i}^{j-1,j}$ is a loss incurred on making single step transitions in both sequences. These losses are given by

$$L_{i-1,i} = \kappa^{(i)}, \qquad L^{j-1,j} = \kappa^{(d)}, \quad (7)$$

$$L_{i-1,i}^{j-1,j} = \kappa^{(s)}(1 - \delta(w_i, w_j^{\mathtt{ref}})) \quad (8)$$

where $\kappa^{(i)}$, $\kappa^{(d)}$ and $\kappa^{(s)}$ are the costs of insertion, deletion and substitution errors. Backtracking along the path with the smallest loss enables each hypothesised word to be marked as either correct or incorrect and thus obtain reference values.

There are two primary modes for evaluating confidence predictors: intrinsic and extrinsic. The intrinsic evaluation assesses confidence scores themselves, whereas the extrinsic evaluation assesses their usefulness in external applications. A number of intrinsic criteria have been proposed, such as normalised cross-entropy (NCE) [63]

$$\text{NCE}(\boldsymbol{c}, \boldsymbol{c}^*) = \frac{H(P_c \cdot \mathbf{1}, \boldsymbol{c}^*) - H(\boldsymbol{c}, \boldsymbol{c}^*)}{H(P_c \cdot \mathbf{1}, \boldsymbol{c}^*)} \quad (9)$$

which provides a relative measure of gain in cross-entropy compared to a baseline that randomly predicts correct confidence with probability $P_c = \frac{1}{T}\sum_{t=1}^{T} c_t^*$ (the average number of correctly transcribed words). It is possible to have positive (better than baseline) and negative (worse than baseline) NCE values. NCE values, however, obfuscate where any gain in performance comes from. An easier to interpret metric can be obtained by choosing a threshold ρ such that any score above that becomes correct and incorrect otherwise. The performance of both the confidence scores and the choice of threshold can be assessed using the standard outcomes of binary detection (true/false positives/negatives) or their derivatives (e.g. accuracy [9], precision, recall, rates). By varying threshold ρ, it is possible to plot either a receiver operating (ROC) or precision and recall (PR) curve respectively, which are useful when deciding an appropriate operating point to use. It is also possible to compute areas under those curves (AUC) to provide a single measure of prediction performance [15], [64].

In order to assess how well confidence scores are calibrated it is common to use reliability diagrams [33]. A reliability diagram is a plot of predicted confidence scores against true confidence scores across the full range of confidence scores. Such plots are created by partitioning predicted confidence scores into bins (e.g. 0-0.2, 0.2-0.35, ..., 0.9-1.0) and plotting the average predicted confidence score against the average true confidence score. A confidence estimation model would be considered calibrated if these values are the same across the full range of confidence scores.

IV. Lattice Context Modelling

Most speech recognisers provide a significantly richer output than the most likely transcription. For example, lattices have one start and one end node associated with the beginning and end of speech and a large number of intermediate nodes that serve as both source and target for one or more arcs. When multiple arcs are connected to the same node then decisions need to be made about how to propagate information forward. Sequence models, such as those described in Section III, cannot directly handle those structures.

This section describes two approaches that enable features to be derived from, and confidences predicted for, all alternative transcriptions and the underlying words. The first is based on lattice recurrent networks where lattice paths are modelled using recurrent network. The second approach uses attention mechanisms over the complete lattice structure.

A. Lattice Recurrent Networks

The key issue to extending RNNs from simple sequences in equation (4) to handling lattices is to address the problem that multiple incoming arcs to a particular arc are present in lattices and CNs. One solution, and the one adopted in this work, is to make use of an attention mechanism to combine all information directly available to a given arc

$$\boldsymbol{h}_{\overrightarrow{\mathcal{N}}_i^{(1)}} = \sum_{j \in \overrightarrow{\mathcal{N}}_i^{(1)}} \alpha_{i,j} \boldsymbol{h}_j \quad (10)$$

before propagating it to any connecting arc

$$\boldsymbol{h}_i = \boldsymbol{\phi}(\boldsymbol{h}_{\overrightarrow{\mathcal{N}}_i^{(1)}}, \boldsymbol{x}_i) \quad (11)$$

where the set of arcs directly preceding arc $\boldsymbol{e}_i$ is denoted by $\overrightarrow{\mathcal{N}}_i^{(1)}$, $\alpha_{i,j}$ is an attention weight associated with arcs $\boldsymbol{e}_i$ and $\boldsymbol{e}_j$, $\boldsymbol{h}_i$ is a history state associated with arc $\boldsymbol{e}_i$. As with RNNs and FNNs before that, the confidence score c_i can be predicted by learning a non-linear transformation of history state $\boldsymbol{h}_i$. Figure 3 provides an illustration of dependencies between features, history states and confidence predictions. Similar to RNNs, it is possible to extend this approach to modelling future information. Such bi-directional lattice RNNs will be examined in Section V.

Attention weights play a critical role in deciding what information will be taken into account. To ensure that weights are non-negative and sum to one, a softmax normalisation

$$\alpha_{i,j} = \frac{\exp(z_{i,j})}{\sum_{j \in \mathcal{N}_i} \exp(z_{i,j})} \quad (12)$$

is applied to the unnormalised weights or energies $z_{i,j}$. The energy computation is often discussed in information retrieval

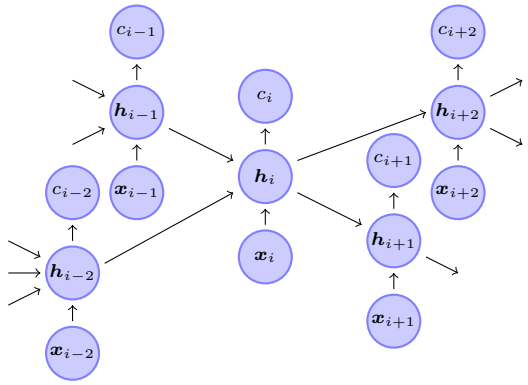

Fig. 3: An example of lattice recurrent NN

terms of queries $\boldsymbol{q}_{i,j}$ and keys $\boldsymbol{k}_{i,j}$, which are used to decide which weight to apply to values, such as states $\boldsymbol{h}_j$ or features $\boldsymbol{x}_j$. A number of approaches have been proposed for computing energies. For instance, additive attention energies can be computed by [14]

$$z_{i,j} = \phi(\boldsymbol{w}^{(z)\mathsf{T}}\boldsymbol{\phi}(\boldsymbol{W}^{(k)}\boldsymbol{k}_{i,j} + \boldsymbol{W}^{(q)}\boldsymbol{q}_{i,j})) \tag{13}$$

as well as [65]

$$z_{i,j} = \phi(\boldsymbol{w}^{(z)\mathsf{T}}\boldsymbol{\phi}(\boldsymbol{W}^{(kq)}[\boldsymbol{k}_{i,j}^{\mathsf{T}} \;\; \boldsymbol{q}_{i,j}^{\mathsf{T}}]^{\mathsf{T}})) \tag{14}$$

where different choices of non-linearities ϕ and $\boldsymbol{\phi}$ provide for more options. On the other hand, multiplicative attention [65]

$$z_{i,j} = \boldsymbol{k}_{i,j}^{\mathsf{T}}\boldsymbol{W}^{(kq)}\boldsymbol{q}_{i,j} \tag{15}$$

includes scaled dot-product [60] and self-attention [60] as special cases. It is also possible to concatenate outputs from multiple (not necessarily the same) attention mechanisms, or heads, to extract diverse kinds of information [60]. This approach will be exploited in Section V to combine different types of contextual information.

To find useful information, the attention mechanism relies on the key to provide a snapshot of the information available and the query to express what is being searched. There are numerous options possible for choosing both. For instance, when deciding if an incoming arc is useful for confidence prediction it is reasonable to use an arc's state as the query

$$\boldsymbol{q}_{i,j} = \begin{bmatrix}\boldsymbol{h}_j\end{bmatrix} \tag{16}$$

and distributional information about word posterior probabilities as the key

$$\boldsymbol{k}_{i,j} = \begin{bmatrix}p_j & \mu_i & \sigma_i\end{bmatrix}^{\mathsf{T}} \tag{17}$$

where μ_i and σ_i are the mean and standard deviation of word posterior probabilities of all incoming arcs. This combination of keys and queries provides the attention mechanism with the content (query) and impact (key) based information.

B. Attention Mechanisms

The recurrent method makes use of an attention mechanism to combine information from neighbouring states. Each of the combined states in turn relies on the attention mechanism to extract information from their respective, direct, neighbourhoods. An alternative approach is to bypass such a repeated process and apply an attention mechanism directly over a large enough neighbourhood[4]

$$\boldsymbol{h}_{\mathcal{N}_i} = \sum_{j\in\mathcal{N}_i} \alpha_{i,j}\boldsymbol{x}_j \tag{18}$$

where $\mathcal{N}_i$ is a set of sequence or graph elements that may be useful for predicting the confidence of the i-th element. Figure 4 shows a simple example of the proposed attention models that makes use of directly connected left neighbours to extract information. In contrast to the recurrent models

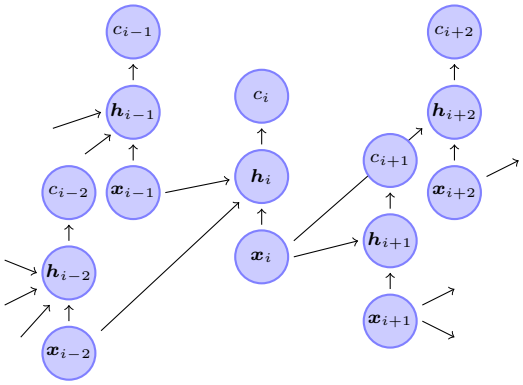

Fig. 4: An example of lattice attention NN

illustrated in Figure 3, attention models, such as in Figure 4, can be efficiently trained. Furthermore, attention models enable efficient computation of confidence scores for a subset of arcs, which may be useful in information retrieval and other applications.

In addition to direct left and right neighbours, there are other options for choosing neighbouring arcs. These include left and right reachable arcs similar to the recurrent method discussed in this section and time-overlapped arcs similar to CNs. It is also possible to extend the notion of neighbouring arcs to include all or some of the arcs from complementary graphs that can be produced using a number of approaches, such as alternative acoustic models, language models and likelihood scales. The flexibility offered by an attention mechanism makes it easy to incorporate other kinds of information, such as topology. When arcs other than direct neighbours are being combined, the simplest example of topological information that can be incorporated into the keys in equation (17) are distances $d_{i,j}$ between the arcs as expressed in terms of the number of arcs [28], binary or probabilistic connectivity masks [29] or time, which can generalise to sets of graphs. Many of these options will be examined in Section V.

C. Network Parameter Training

The recurrent and attention models proposed in this section can be trained by minimising binary cross-entropy with respect

[4]Although attention mechanisms have previously appeared in confidence prediction models [66]–[68], their use have been constrained to extracting information from encoder/decoder states of ASR systems and hypothesised one-best sequences.

to the reference confidence scores. In the simplest case, only those confidence scores that are linked with the most likely transcripts can be predicted. Such an approach eliminates the need to obtain reference confidence values for a large number of competing transcripts but may be suboptimal if confidence scores linked with them are required. Alternatively, it is possible to predict confidence scores for all arcs of confusion networks or lattices provided reference values are available.

Confusion networks (CN): Given a hypothesis CN $\boldsymbol{\mathcal{C}}$ and a reference word sequence $\boldsymbol{w}^{\mathtt{ref}}$, the Levenshtein algorithm can be adopted to mark CN arcs with substitution loss set to

$$L_{i-1,i}^{j-1,j} = \kappa^{(s)}(1 - P(w_j^{\mathtt{ref}}|\mathcal{C}_i, \boldsymbol{O})) \quad (19)$$

When references are provided in the form of CNs, the posterior probability above is replaced by

$$P(\mathcal{C}_j^{\mathtt{ref}}|\mathcal{C}_i, \boldsymbol{O}) = \sum_{w_j^{\mathtt{ref}} \in \mathcal{C}_j^{\mathtt{ref}}} P(w_j^{\mathtt{ref}}|\mathcal{C}_i, \boldsymbol{O}) P(w_j^{\mathtt{ref}}|\mathcal{C}_j^{\mathtt{ref}}, \boldsymbol{O}) \quad (20)$$

to yield CN alignment or combination (CNC) [22]. Similar to sequences in Section III, references for CNs can be obtained by backtracking along the path with the smallest loss.

Lattices: Marking lattice arcs with reference confidence scores is significantly more challenging. Instead, approximate marking schemes based on time overlap can be adopted [69]. Given a hypothesis arc $\boldsymbol{e}_i$ and reference arc $\boldsymbol{e}_j^*$ with start times $t_i^{(s)}$ and $t_j^{(s)}$, end times $t_i^{(e)}$ and $t_j^{(e)}$, and identical word labels, the time-overlap can be estimated by

$$\nu_{i,j} = \max \left\{ 0, \frac{|\min(t_j^{(e)}, t_i^{(e)})| - |\max(t_j^{(s)}, t_i^{(s)})|}{|\max(t_j^{(e)}, t_i^{(e)})| - |\min(t_j^{(s)}, t_i^{(s)})|} \right\} \quad (21)$$

Given a fixed threshold ν, any hypothesis arc $\boldsymbol{e}_i$ with $\nu_{i,j} \geq \nu$ will be marked as correct with respect to the reference arc $\boldsymbol{e}_j^*$.

V. Experiments

This section describes experiments that were conducted with recurrent and attention-based neural network approaches for predicting confidence scores for the most likely transcriptions generated by Cambridge University submissions to IARPA Babel competitions. OpenKWS [70] and their successor OpenCLIR [71] public competitions challenge participants to develop robust speech recognisers for limited resource languages to support information retrieval tasks. Word error rates for those languages commonly range between 20-60% [72] and necessitate the use of error mitigation approaches, such as confidence scores, to achieve high performance.

A. Setup

Most of the evaluation was conducted using the Georgian full language pack (FLP), which consists of approximately 40 hours of transcribed training data for building speech recognisers and 10 hours of development data for testing them. All speech data are telephone conversations recorded at 8kHz, mostly over mobile phone networks. The speech recogniser is a complex acoustic model that combines 4 diverse acoustic models [72]. The diversity is accomplished through the use of different model architectures (hybrid and tandem) and different multilingual bottleneck features. The multilingual features were estimated by IBM and RWTH Aachen on a collection of 28 languages packs released by IARPA and LDC. All acoustic models are based on graphemic lexica which were derived using automatic approaches [73]. The language model used in this article is a simple trigram language model estimated on training data transcripts and web data. The speech recogniser was used to produce a set of lattices using a default grammar scale factor (20) and 4 perturbed factors (12, 16, 24 and 28). The default factor was selected based on a broad range of other IARPA Babel languages. Lattices were converted into confusion networks (CN) using confusion network decoding [7]. The most likely transcripts were obtained from the output of CN decoding that corresponds to the default grammar scale factor.

TABLE I: Baselines for confidence estimation approaches

Model	NCE	AUC_{PR}	
		$\text{AUC}_{\text{PR}}^{(0)}$	$\text{AUC}_{\text{PR}}^{(1)}$
random	0.0	0.3177	0.6823
posterior	-0.1978	0.7112	0.9081
decision tree	0.2755	0.7112	0.9081

The available development data were partitioned into a training, development and evaluation set with a ratio of 8:1:1 for training, validating and testing confidence estimation schemes. The most likely transcription of the evaluation set contains 6063 words of which 4137 or 68.2% words are correctly predicted. Three baseline schemes were examined: 1) a random classifier, 2) word posterior probabilities, 3) a decision tree. Table I provides a snapshot of their performance on the evaluation set. The NCE for posterior probabilities (-0.1978) is negative, which suggests that uncalibrated posteriors are less informative than the random classifier that predicts confidence of 1 with probability $P_c = 0.682$. The decision tree, as expected, improves calibration of posterior probabilities (0.2755). Areas under the precision-recall curves, where $\text{AUC}_{\text{PR}}^{(0)}$ treats incorrect words as positives and $\text{AUC}_{\text{PR}}^{(1)}$ treats incorrect words as negatives, provide additional information. Unlike NCEs, AUCs clearly show that given an appropriate threshold to map posterior probabilities to either 0 or 1 a significantly better performance than the random classifier can be achieved. Furthermore, it appears that determining which predictions are incorrect is a significantly harder problem. Given a smaller number of incorrect predictions in the most likely transcriptions, it will be more challenging to improve accuracy of determining incorrect predictions. In common with other work in this area, all results are initially presented in terms of NCE. Section V-D will discuss all major results in terms of other performance criteria.

B. Sequences

The first set of experiments examined the possibility of learning accurate confidence scores using information available only within the most likely transcriptions. Both recurrent and attention-based models were examined.

TABLE II: Recurrent sequence models

Features		NCE
Word	embedding	0.0358
	+duration	0.0541
	+posterior	0.2765
	+decision tree	0.2911
Grapheme	+embedding	0.2936
	+duration	0.2944
	+encoder	0.2978

The recurrent models are bidirectional and make use of 128-dimensional long short-term memory (LSTM) units [52]. A range of word and sub-word (grapheme) features have been examined. Word features comprise 50-dimensional `fastText` [74] word embeddings and 1-dimensional duration, uncalibrated and decision tree calibrated posterior probabilities. The first horizontal block in Table II shows how NCE changes as more word features are used. Simple features, such as word embeddings and duration, offer a limited gain over the random baseline. The use of word posterior probabilities, unsurprisingly, brings a very large gain in NCE. Despite having access to word embeddings and duration information the use of more powerful BiRNNs has led to a small improvement over decision tree calibration ($0.2755 \rightarrow 0.2765$). However, when BiRNNs make use of calibrated word posteriors instead then the gain over decision tree calibration becomes significant ($0.2755 \rightarrow 0.2911$). A similar observation was made in the context of E2E ASR systems [46].

Grapheme features provide one of many possible options for extending available features and comprise 4-dimensional `word2vec` [61] grapheme embeddings, 1-dimensional duration and 10-dimensional encoder output based on bi-directional gated recurrent units. Grapheme features were combined with word features by learning an attention mechanism to map a variable number of grapheme features to fixed length as described in [62]. The second horizontal block in Table II shows how NCE changes as richer grapheme features are extracted. Overall, the use of grapheme features provides a significant increase in NCE ($0.2911 \rightarrow 0.2978$). Due to the increased complexity of learning grapheme encoders to extract features and attention mechanisms to combine them, the rest of this section will focus on word features only.

The attention models can make use of one or more attention mechanisms to combine information across the most likely word sequence. There are a number of possible context spans to choose from: one or more left neighbours, one or more right neighbours, left reachable words, right reachable words, all reachable words. In common with other work in this area, the positional information has been encoded into the keys using discrete distances equal to the number of arcs that need to be traversed to connect any two arcs with positive distances representing following arcs and negative distances representing preceding arcs. All attention models in this article transform combined features using three 64-dimensional ReLU layers prior to mapping features to $[0, 1]$ range using a sigmoid non-linearity.

TABLE III: Attention mechanisms for sequences

Attention Mechanisms		NCE
I	II	
—	—	0.2895
left neighbours	right neighbours	0.2908
left-reachable	—	0.2917
left-reachable	right-reachable	0.2920
all reachable	—	0.2919

TABLE IV: Attention mechanism options

(a) Additive heads

Heads	NCE
1	0.2919
2	0.2941
4	0.2949
8	0.2920

(b) Attention type

Type	NCE
additive	0.2919
multiplicative	0.2928
scaled dot product	0.2897

Table III shows how NCE changes as the context of information available to attention models increases from no contextual information (0.2895) to all left and right reachable words (0.2920). It is interesting that the former performance is only slightly worse than the performance of a BiRNN that has access to all past and future words (0.2911 in Table II). Comparing the model that has access to only past information (left-reachable) to the model with both the past (left-reachable) and future (right-reachable) information, it appears that the future information provides only a limited improvement (0.2917 vs. 0.2920). This observation suggests that accurate confidence estimates can be obtained in streaming applications where access to future information may not be possible. The final row shows that the use of dedicated attention heads to incorporate past and future information separately offers little gain over a single attention mechanism that has access to all reachable words.

As discussed in Section IV, there are many possible ways to compute attention weights. The experiments in Table III made use of the additive attention in equation (14), where $\boldsymbol{W}^{(kq)}$ is a 57×57 parameter matrix (53-dimensional features and 4-dimensional keys). Table IV explores (a) multiple additive attention heads and (b) alternatives to additive attention, such as multiplicative and scaled dot product attention. The NCE results suggest that both directions can bring substantial gains. For simplicity the following sections will focus on the additive attention mechanism with one head.

C. Confusion networks

The recurrent and attention models examined so far have been constrained to extract information from most likely sequences. The second set of experiments examined incorporating additional contextual information from alternative transcriptions. The set of CNs that yielded the most likely transcriptions was used to train recurrent and attention models. Table V provides a side-by-side comparison between recurrent and attention models. As discussed in Section IV graph-based models offer an opportunity to evaluate and optimise

TABLE V: Feature extraction and loss computation based on confusion networks

Source CN Arcs		NCE	
Features	Loss	Recurrent	Attention
1-best arcs	1-best arcs	0.2911	0.2919
all arcs	1-best arcs	0.2931	0.2925
all arcs	all arcs	0.2934	0.2948

TABLE VI: Confusion network attention options

(a) Distance types

Distance	NCE
arc	0.2948
time	0.2962
arc and time	0.2965

(b) Attention mechanisms

Attention Mechanisms		NCE
I	II	
reachable	—	0.2962
all	—	0.2967
reachable	time-overlapped	0.3001

loss over a subset of arcs, e.g. arcs that form the most likely transcriptions. The NCE results in Table V suggest that attention models benefit significantly more from optimising loss on all arcs than recurrent models.

The positional information has so far been represented by discrete arc-based distances. As mentioned in Section IV-B it is possible to measure distances using other approaches, such as time, that provide a more nuanced distance estimate. Using duration information available to each CN arc enables a continuous estimate of distance to be obtained. Table VI (a) shows that time-based distances offer a clear advantage over arc-based distances. Furthermore, making use of both these distances as expected yields a marginal gain in NCE performance.

The set of reachable arcs used by recurrent and attention models to make a confidence prediction excludes competing arcs. For instance, word AN in Figure 2 is just one of four possible words within that CN bin. It is expected that the knowledge of competing words should help to predict confidence scores more accurately. To verify this hypothesis, a second attention head was introduced where the context span was limited to arcs present only within the respective CN bin. Table VI (b) shows that merging competing, time-overlapped, arcs into the set of reachable arcs yields small gains in NCE ($0.2962 \rightarrow 0.2967$). However, larger gains can be obtained if competing arcs are modelled by a separate attention mechanism ($0.2962 \rightarrow 0.3001$).

The final experiment examined incorporating information from a set of CNs. As mentioned at the beginning of this section, the diversity of CNs in this article was achieved by varying language model scale. Table VII shows that a simple approach of aggregating all arcs across 5 CNs does not yield any gain in NCE performance ($0.2967 \rightarrow 0.2962$). On the other hand, a large gain in NCE performance is observed when a separate attention mechanism is introduced to model competing words across all 5 CNs ($0.2967 \rightarrow 0.3035$).

TABLE VII: Attention mechanisms for confusion networks

Attention Mechanisms		NCE
I	II	
all arcs (1 CN)	—	0.2967
all arcs (5 CN)	—	0.2962
all arcs (1 CN)	time-overlapped (5 CN)	0.3035

TABLE VIII: Performance summary of baseline and neural network confidence estimation approaches

Context	Model	NCE	AUC	
			$\text{AUC}_{\text{PR}}^{(0)}$	$\text{AUC}_{\text{PR}}^{(1)}$
1-best	decision tree	0.2755	0.7112	0.9081
	recurrent	0.2911	0.7194	0.9178
	attention	0.2949	0.7209	0.9171
CN	recurrent	0.2934	0.7185	0.9197
	attention	0.3001	0.7312	0.9189
5 CNs	attention	0.3035	0.7340	0.9205

D. Detailed performance analysis

As discussed in Section IV, the use of attention models provides a highly flexible framework for incorporating information across a wide range of commonly used representations. The current section has so far presented empirical evidence to support those claims based on the NCE criterion. Table VIII provides a summary of NCE values of all major confidence estimation models examined in this article. Although attention models yield significant gains in NCE performance over the decision tree approach ($0.2755 \rightarrow 0.3035$), care is required using only NCE criterion. Table VIII also compares performance in terms of the area under precision-recall curves, which confirm that advanced neural network approaches enable more incorrect ($\text{AUC}_{\text{PR}}^{(0)}$) and correct ($\text{AUC}_{\text{PR}}^{(1)}$) arcs to be correctly classified compared to the decision tree-based approach. As was expected classifying incorrect arcs as incorrect appears to be more challenging than classifying correct arcs as correct. Furthermore, it appears that the neural network approaches improve more in the former case which will benefit applications focused on detecting errors.

Reliability diagrams in Figure 5 provide additional confirmation by showing a significantly better calibration of neural network approaches over the decision tree for low values (0-0.3) of predicted confidence scores. These diagrams provide a clear illustration of poor calibration offered by word posterior probabilities and the impact that simple, such as decision trees, and complex, such as attention models, schemes have on calibration. The neural network approaches offer a significantly better consistency but visibly under/over predict confidence near 0.3 and 0.8.

E. Other limited resource languages

Georgian is one of dozens of limited resource languages examined in the IARPA Babel and, its successor, MATERIAL

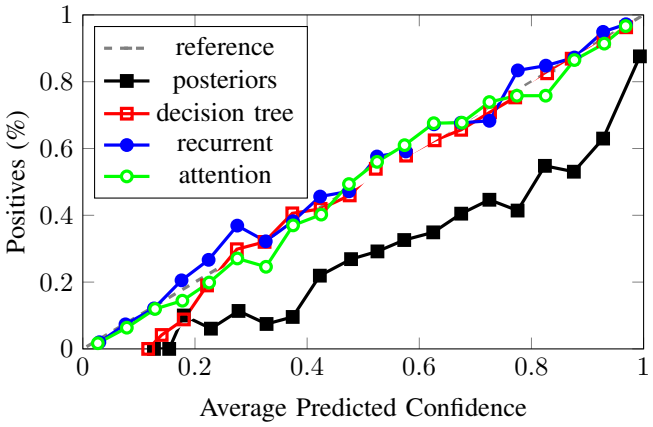

Fig. 5: Reliability diagrams for selected confidence estimation approaches

TABLE IX: Performance summary of baseline and neural network confidence estimation approaches on 4 diverse languages

Language	WER (%)	Model	NCE	AUC	
				$AUC_{PR}^{(0)}$	$AUC_{PR}^{(1)}$
Georgian	38.4	decision tree	0.2755	0.7112	0.9081
		attention	0.3001	0.7312	0.9189
Swahili	44.7	decision tree	0.2580	0.7448	0.8739
		attention	0.2772	0.7531	0.8821
Javanese	50.5	decision tree	0.1782	0.6837	0.8255
		attention	0.2483	0.7541	0.8632
Igbo	54.1	decision tree	0.1646	0.7120	0.7864
		attention	0.2000	0.7545	0.8086

programmes. Those languages were carefully selected by the MIT Lincoln Laboratory to provide a representative and diverse sample of languages. Recogniser accuracy for those languages is typically lower than what can be achieved for English and varies in the range of 30-60% WER, which poses a significant challenge in developing accurate solutions that utilise ASR technology. Three languages were selected from that range: Swahili, Javanese and Igbo (see [72] for setup). As shown in Table IX, WERs for these languages are substantially higher than for Georgian. Confidence score quality, as measured by the decision tree calibrated NCE values and $AUC_{PR}^{(1)}$, appears to be negatively correlated with WER. On the other hand, the correlation to $AUC_{PR}^{(0)}$ is weaker. The decision tree yields lower than expected NCE values for the two most challenging languages. The attention models bring gains over decision trees for all languages. In particular, substantially larger gains for the two most challenging languages address the limitations of simpler decision trees. A similar picture can be observed in terms of AUC values, where substantially larger gains are observed for more challenging languages. These observations suggest that confidence estimation approaches may provide a substantial performance improvement in situations where ASR systems exhibit poor performance.

VI. Conclusion

Confidence scores play an important role in the development and adoption of speech and language technology, and their applications. A wide range of approaches have been developed over the years with the aim to improve over the simplest form of confidence scores – word posterior probabilities. This article argues that context plays a key role in the assessment of ASR system prediction accuracy, and shows how neural network approaches, such as RNNs and attention, can be extended to combine diverse types (word and subword level) of information from varied sources (sequences, graphs and sets of graphs). In particular, the article shows how to devise high-performing recurrent and attention models over complex sources, such as confusion networks, for confidence prediction. Experimental validation was performed using the IARPA OpenKWS 2016 challenge Georgian language. The experimental results show that the proposed approaches provide higher accuracy and better consistency than word posteriors and simple calibration schemes across the full range of confidence scores. These findings were further corroborated on three other challenging IARPA Babel programme languages.

References

[1] M. S. Seigel and P. C. Woodland, "Detecting deletions in ASR output," in *ICASSP*, 2014, pp. 2302–2306.
[2] A. Ragni, Q. Li, M. J. F. Gales, and Y. Wang, "Confidence estimation and deletion prediction using bidirectional recurrent neural networks," in *SLT*, 2018, pp. 204–211.
[3] G. Zavaliagkos, M. Siu, T. Colthurst, and J. Billa, "Using untranscribed training data to improve performance," in *ICSLP*, 1998, p. 1007.
[4] M. Pitz, F. Wessel, and H. Ney, "Improved MLLR speaker adaptation using confidence measures for conversational speech recognition," in *ICSLP*, vol. 4, 2000, pp. 548–551.
[5] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recogniser output voting error reduction (ROVER)," in *ASRU*, 1997, pp. 347–352.
[6] C. Neti, S. Roukos, and E. Eide, "Word-based confidence measures as a guide for stack search in speech recognition," in *ICASSP*, 1997.
[7] G. Evermann and P. C. Woodland, "Large vocabulary decoding and confidence estimation using word posterior probabilities," in *ICASSP*, 2000, pp. 1655–1658.
[8] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus in speech recognition: word error minimization and other applications of confusion networks," *Comp Speech & Language*, vol. 14, no. 4, pp. 373–400, 2000.
[9] F. Wessel, R. Schlüter, K. Macherey, and H. Ney, "Confidence measures for large vocabulary continuous speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 3, pp. 288–298, 2001.
[10] M. S. Seigel and P. C. Woodland, "Combining information sources for confidence estimation with CRF models," in *INTERSPEECH*, 2011.
[11] K. Kalgaonkar, C. Liu, Y. Gong, and K. Yao, "Estimating confidence scores on ASR results using recurrent neural networks," in *ICASSP*, 2015, pp. 4999–5003.
[12] A. Ogawa and T. Hori, "Asr error detection and recognition rate estimation using deep bidirectional recurrent neural networks," in *ICASSP*, 2015, pp. 4370–4374.
[13] A. Ogawa and T. Hori, "Error detection and accuracy estimation in automatic speech recognition using deep bidirectional recurrent neural networks," *Speech Communication*, vol. 89, pp. 70–83, 2017.
[14] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *ICLR*, 2015.
[15] Q. Li, P. M. Ness, A. Ragni, and M. J. F. Gales, "Bi-directional lattice recurrent neural networks for confidence estimation," in *ICASSP*, 2019.
[16] T. Kemp and T. Schaaf, "Estimating confidence using word lattices," in *EUROSPEECH*, 1997, pp. 827–830.
[17] L. Gillick, Y. Ito, and J. Young, "A probabilistic approach to confidence estimation and evaluation," in *ICASSP*, vol. 2, 1997, pp. 879–882.
[18] F. Ladhak, A. Gandhe, M. Dreyer, L. Mathias, A. Rastrow, and B. Hoffmeister, "LatticeRNN: Recurrent neural networks over lattices," in *INTERSPEECH*, 2016, pp. 695–699.

[19] J. Su, Z. Tan, D. Xiong, R. Ji, X. Shi, and Y. Liu, "Lattice-based recurrent neural network encoders for neural machine translation," in *AAAI*, 2017, pp. 3302–3308.

[20] J. Buckman and G. Neubig, "Neural lattice language models," *Transactions of the ACL*, vol. 6, pp. 529–541, 2018.

[21] Y. Zhang and J. Yang, "Chinese NER using lattice LSTM," in *ACL*, 2018, pp. 1554–1564.

[22] G. Evermann and P. C. Woodland, "Posterior probability decoding, confidence estimation and system combination," in *Speech Transcription Workshop*, 2000.

[23] H. Printz and P. A. Olsen, "Theory and practice of acoustic confusability," *Computer Speech and Language*, vol. 16, pp. 131–164, 2002.

[24] R. Bippus, A. Fischer, and V. Stahl, "Domain adaptation for robust automatic speech recognition in car environments," in *Eurospeech*, 1999.

[25] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," in *ACL*, 2015.

[26] Y. Li, R. Zemel, M. Brockschmidt, and D. Tarlow, "Gated graph sequence neural networks," in *ICLR*, 2016.

[27] M. Sperber, G. Neubig, J. Niehues, and A. Waibel, "Neural lattice-to-sequence models for uncertain inputs," in *EMNLP*, 2017.

[28] P. Zhang, B. Chen, N. Ge, and K. Fan, "Lattice transformer for speech translation," in *ACL*, 2019, pp. 6475–6484.

[29] M. Sperber, G. Neubig, N.-Q. Pham, and A. Waibel, "Self-attentional models for lattice inputs," in *ACL*, 2019, pp. 1185–1197.

[30] A. Graves, S. Fernandez, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *ICML*, 2006, pp. 369–376.

[31] A. Graves, "Sequence transduction with recurrent neural networks," *arXiv:1211.3711*, 2012.

[32] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *ICASSP*, 2016, pp. 4960–4964.

[33] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *ICML*, 2017, pp. 1321–1330.

[34] M. Kull, M. Perello-Nieto, M. Kängsepp, T. S. Filho, H. Song, and P. Flach, "Beyond temperature scaling: Obtaining well-calibrated multiclass probabilities with Dirichlet calibration," in *NeurIPS*, 2019.

[35] M. Zapotoczny, P. Pietrzak, A. Lancucki, and J. Chorowski, "Lattice generation in attention-based speech recognition models," in *Interspeech*, 2019, pp. 2225–2229.

[36] R. Prabhavalkar, Y. He, D. Rybach, S. Campbell, A. Narayanan, T. Strohman, and T. N. Sainath, "Less is more: Improved RNN-T decoding using limited label context and path merging," in *ICASSP*, 2021, pp. 5659–5663.

[37] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[38] H. Zen, K. Tokuda, and T. Kitamura, "Reformulating the HMM as a trajectory model by imposing explicit relationships between static and dynamic feature vector sequences," *Computer Speech & Language*, vol. 21, no. 1, pp. 153–173, 2007.

[39] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *NIPS*, 2017.

[40] G.-L. Tran, E. V. Bonilla, J. Cunningham, P. Michiardi, and M. Filippone, "Calibrating deep convolutional Gaussian processes," in *AISTATS*, 2019, pp. 1554–1563.

[41] A. Kumar, S. Sarawagi, and U. Jain, "Trainable calibration measures for neural networks from kernel mean embeddings," in *ICML*, 2018.

[42] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *ICLR*, 2018.

[43] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Advances in Large Margin Classifiers*, vol. 10, no. 3, 1999.

[44] A. Kumar, P. S. Liang, and T. Ma, "Verified uncertainty calibration," in *NeurIPS*, 2019, pp. 3792–3803.

[45] M. Weintraub, F. Beaufays, Z. Rivlin, Y. Konig, and A. Stolcke, "Neural-network based measures of confidence for word recognition," in *ICASSP*, vol. 2, 1997, pp. 887–890.

[46] Q. Li, D. Qiu, Y. Zhang, B. Li, Y. He, P. C. Woodland, L. Cao, and T. Strohman, "Confidence estimation for attention-based sequence-to-sequence models for speech recognition," in *ICASSP*, 2021.

[47] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[48] A. Gunawardana, M. Mahajan, A. Acero, and J. C. Platt, "Hidden conditional random fields for phone classification," in *INTERSPEECH*, 2005, pp. 1117–1120.

[49] G. Zweig and P. Nguyen, "A segmental CRF approach to large vocabulary continuous speech recognition," in *ASRU*, 2009, pp. 152–157.

[50] A. Ragni and M. J. F. Gales, "Derivative kernels for noise robust ASR," in *ASRU*, 2011, pp. 119–124.

[51] C. Sutton and A. McCallum, *An Introduction to Conditional Random Fields*. NOW Publishers, 2011.

[52] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[53] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *EMNLP*, 2014, pp. 1724–1734.

[54] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Tran on Sig Proc*, vol. 45, no. 11, pp. 2673–2681, 1997.

[55] M. A. Del-Agua, A. Giménez, A. Sanchis, J. Civera, and A. Juan, "Speaker-adapted confidence measures for ASR using deep bidirectional recurrent neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 7, pp. 1198–1206, 2018.

[56] Q. Li, Y. Zhang, B. Li, L. Cao, and P. Woodland, "Residual energy-based models for end-to-end speech recognition," in *Interspeech*, 2021.

[57] E. Eide, H. Gish, P. Jeanrenaud, and A. Mielke, "Understanding and improving speech recognition performance through the use of diagnostic tools," in *ICASSP*, vol. 1, 1995, pp. 221–224.

[58] T. Jaakkola and D. D. Hausser, "Exploiting generative models in discriminative classifiers," in *NIPS*, 1999, pp. 487–493.

[59] M. Layton, "Kernel methods for classifying variable length data," Ph.D. dissertation, University of Cambridge, 2006.

[60] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *NIPS*, 2017, pp. 6000–6010.

[61] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *NIPS*, vol. 26, 2013, pp. 3111–3119.

[62] A. Kastanos, A. Ragni, and M. J. F. Gales, "Confidence estimation for black box automatic speech recognition systems using lattice recurrent neural networks," in *ICASSP*, 2020, pp. 6329–6333.

[63] S. Cox and R. Rose, "Confidence measures for the SWITCHBOARD database," in *ICASSP*, vol. 1, 1996, pp. 511–514.

[64] D. Yu, J. Li, and L. Deng, "Calibration of confidence measures in speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 8, pp. 2461–2473, 2010.

[65] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *EMNLP*, 2015.

[66] A. Kumar, S. Singh, D. Gowda, A. Garg, S. Singh, and C. Kim, "Utterance confidence measure for end-to-end speech recognition with applications to distributed speech recognition scenarios," in *INTERSPEECH*, 2020.

[67] D. Qiu, Q. Li, Y. He, Y. Zhang, B. Li, L. Cao, R. Prabhavalkar, D. Bhatia, W. Li, K. Hu, T. Sainath, and I. McGraw, "Learning word-level confidence for subword end-to-end ASR," in *ICASSP*, 2021.

[68] D. Qiu, Y. He, Q. Li, Y. Zhang, L. Cao, and I. McGraw, "Multi-task learning for end-to-end ASR word and utterance confidence with deletion prediction," in *Interspeech*, 2021.

[69] D. Povey, "Discriminative training for large vocabulary speech recognition," Ph.D. dissertation, University of Cambridge, 2004.

[70] 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 *SLTU*, 2014, pp. 16–23.

[71] C. Rubino, "The effect of linguistic parameters in CLIR performance," in *Workshop on Cross-Language Search and Summarization of Text and Speech*, 2020, pp. 1–6.

[72] A. Ragni, C. Wu, M. J. F. Gales, J. Vasilakes, and K. M. Knill, "Stimulated training for automatic speech recognition and keyword search in limited resource conditions," in *ICASSP*, 2017, pp. 4830–4834.

[73] M. J. F. Gales, K. M. Knill, and A. Ragni, "Unicode-based graphemic systems for limited resource languages," in *ICASSP*, 2015.

[74] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *arXiv:1607.04606*, 2016.

Anton Ragni Anton Ragni received B.Eng. and M.Eng. degrees in Information Technology from the University of Tartu, Estonia, in 2005 and 2007 respectively. From 2005 to 2008 he underwent graduate training at the Nordic Graduate School of Language Technology, Sweden, and from 2007 to 2008 was an intern in the Speech Technology Group, Toshiba Research Europe Limited, UK. In 2008 Anton commenced his doctoral studies at the University of Cambridge, UK, and completed his PhD degree in Automatic Speech Recognition in 2013. He received "Best Student Paper Award" at IEEE Workshop on Automatic Speech Recognition (ASR) and Understanding for his paper "Generative kernels for noise robust ASR" co-authored with Mark J.F. Gales in 2011. From 2013 to 2018 Anton was a Research Associate and from 2018-2019 he was a Senior Research Associate in Speech Processing at the University of Cambridge working on the IARPA BABEL and MATERIAL projects, and at the Institute for Automated Language Teaching and Assessment (ALTA). In 2019 he was appointed as a Lecturer in Speech and Language Technologies at the University of Sheffield, UK. He is a Co-Director of Enhanced Speech Technology Ltd. Anton is a Member of IEEE and ISCA and since 2016 he has been an officer of ISCA Special Interest Group on Machine Learning in Speech and Language Processing (MLSLP).

Mark J. F. Gales studied for the B.A. in Electrical and Information Sciences at the University of Cambridge from 1985-88. Following graduation he worked as a consultant at Roke Manor Research Ltd. In 1991 he took up a position as a Research Associate in the Speech, Vision and Robotics Group in the Engineering Department at Cambridge University. In 1995 he completed his doctoral thesis: Model-Based Techniques for Robust Speech Recognition supervised by Professor Steve Young. From 1995-1997 he was a Research Fellow at Emmanuel College, Cambridge. He was then a Research Staff Member in the Speech group at the IBM T.J.Watson Research Center until 1999 when he returned to Cambridge University Engineering Department as a University Lecturer. He was appointed Reader in Information Engineering in 2004. He is currently a Professor of Information Engineering and a College Lecturer and Official Fellow of Emmanuel College. He is a Co-Founder and Co-Director of Enhanced Speech Technology Ltd. Mark Gales is a Fellow of the IEEE and ISCA and was a member of the IEEE Speech and Language Processing Technical Committee (2001-2004 and 2015-2017). He was an associate editor for IEEE Signal Processing Letters from 2008-2011 and IEEE Transactions on Audio Speech and Language Processing from 2009-2013. He is currently on the Editorial Board of Computer Speech and Language. He has been awarded a number of paper awards, including a 1997 IEEE Young Author Paper Award for his paper on Parallel Model Combination and a 2002 IEEE Paper Award for his paper on Semi-Tied Covariance Matrices.

Oliver Rose completed a B.A. and M.Eng. in Electrical and Information Engineering from the University of Cambridge in 2020. Supervised by Dr. Anton Ragni, Prof. Mark Gales and Dr. Kate Knill, his research for his M.Eng. focused on exploring the use of attention based models for the estimation of confidence scores on the output of ASR systems. Currently he works as a computer vision researcher for Oxehealth, a medical technology company focusing on remote patient monitoring.

Katherine M. Knill received a B.Eng. (Jt. Hons.) in Electronic Engineering and Mathematics from the University of Nottingham in 1990, and a PhD in Adaptive Digital Signal Processing from Imperial College, University of London, UK, in 1994, supervised by Prof. Tony Constantinides. She was sponsored on both by Marconi Underwater Systems Ltd, UK. From 1993-1996 she was a Research Associate in the Speech Vision and Robotics Group in the Engineering Department at Cambridge University. In 1997 she joined the Speech R&D Team, Nuance Communications, Menlo Park, USA, becoming Languages Manager in 2000. Kate established a new Speech Technology Group at Toshiba Research Europe Ltd, Cambridge Research Lab, UK, in 2002. As Assistant Managing Director and Speech Technology Group Leader she was responsible for interactive technology, in particular core speech recognition and synthesis R&D and development of European and North American speech products. She returned to Cambridge University Engineering Department as a Senior Research Associate in Spoken Language Processing (SLP) in 2012, working on the IARPA BABEL and MATERIAL projects and at the Institute for Automated Language Teaching and Assessment (ALTA). Kate was appointed a Principal Research Associate in 2019 and is the lead PI for the ALTA SLP Technology Project. She is a Co-Founder and Co-Director of Enhanced Speech Technology Ltd. Kate Knill is a Member of the IEEE, ISCA and IET. She was a member of the IEEE SLTC 2009-2012, an ISCA Board member 2013-2021 and ISCA Secretary 2017-2021. She is currently a member of the IEEE James L. Flanagan Speech and Audio Processing Award committee.

Alexandros Kastanos Alexandros Kastanos received a B.Sc. in Electrical Engineering from the University of the Witwatersrand in 2017 and an M.Phil. in Machine Learning and Machine Intelligence from the University of Cambridge in 2019. His current research interests include low-resource natural language processing and speech recognition as well as applied ethics. In conjunction with his independent research, he works as a machine learning engineer at a leading FoodTech company in London.

Qiujia Li is a PhD student in the Machine Intelligence Laboratory at the University of Cambridge, supervised by Prof. Phil Woodland. He obtained B.A. and M.Eng. degrees in Information and Computer Engineering from the University of Cambridge in 2018. He received the Best Student Paper Awards at IEEE ASRU 2019 for his paper "Integrating source-channel and attention-based sequence-to-sequence models for speech recognition" and at IEEE SLT 2021 for his paper "Discriminative neural clustering for speaker diarisation". His current research interests include speech recognition, confidence estimation and speaker diarisation. He is a student member of IEEE and ISCA.

Preben M. Ness completed the degrees B.A. in Engineering and M.Eng. in Information and Computer Engineering at the University of Cambridge, graduating in 2019. During the summer of 2018 he worked as an Undergraduate Researcher at the Machine Intelligence Laboratory, researching confidence scores in ASR systems. Preben is currently working as an ML Engineer at the Nordic crypto-currency exchange Firi. His research interests include uncertainty estimation on structured data, time-series anomaly detection, and predictive modelling using Graph Neural Networks.