A HIERARCHICAL ATTENTION BASED MODEL FOR OFF-TOPIC SPONTANEOUS SPOKEN RESPONSE DETECTION

Andrey Malinin, Kate Knill and Mark J. F. Gales

University of Cambridge, Department of Engineering,
Trumpington St, Cambridge CB2 1PZ, UK

ABSTRACT

Automatic spoken language assessment and training systems are becoming increasingly popular to handle the growing demand to learn languages. However, current systems often assess only fluency and pronunciation, with limited content-based features being used. This paper examines one particular aspect of content-assessment, off-topic response detection. This is important for deployed systems as it ensures that candidates understood the prompt, and are able to generate an appropriate answer. Previously proposed approaches typically require a set of prompt-response training pairs, which limits flexibility as example responses are required whenever a new test prompt is introduced. Recently, the attention based neural topic model (ATM) was presented, which can assess the relevance of prompt-response pairs regardless of whether the prompt was seen in training. This model uses a bidirectional Recurrent Neural Network (BiRNN) embedding of the prompt combined with an attention mechanism to attend over the hidden states of a BiRNN embedding of the response to compute a fixed-length embedding used to predict relevance. Unfortunately, performance on prompts not seen in the training data is lower than on seen prompts.

Thus, this paper adds the following contributions: several improvements to the ATM are examined; a hierarchical variant of the ATM (HATM) is proposed, which explicitly uses prompt similarity to further improve performance on unseen prompts by interpolating over prompts seen in training data given a prompt of interest via a second attention mechanism; an in-depth analysis of both models is conducted and main failure mode identified. On spontaneous spoken data, taken from BULATS tests, these systems are able to assess relevance to both seen and unseen prompts.

Index Terms: Spoken Language Assessment, Relevance Assessment, Deep Learning

1. INTRODUCTION

A key part of learning a language is learning how to speak fluently and with confidence. The level reached can be assessed through spoken language proficiency tests where candidates are prompted to respond to a series of open-ended questions, such as ”describe a difficult situation at work, why was it difficult?”. Human examiners assess the candidate’s spontaneous speech replies in terms of pronunciation, hesitations/extent, use of grammar and vocabulary, and how coherent their discourse is. The increasing demand for language learning and for practice tests available at any time make the development of automatic assessment systems which can also provide feedback an attractive proposition [1]. Structured features derived from automatic speech recognition (ASR) generated transcriptions of the candidate’s responses are combined with features derived directly from the audio as input to automatic spoken language assessment systems. Such systems currently primarily focus on pronunciation and fluency (both of which are highly correlated with proficiency) with minimal content assessment, e.g. ETS’ *SpeechRater* [2] and Pearson’s *AZELLA* [3]. However, reliable, robust assessment requires the evaluation of the semantic content, construction and relevance of a response to the question prompt. This assumes a response corresponds to the question asked, so such an automatic system must assess if a candidate has given an off-topic response, either due to misunderstanding the question and/or memorizing a response. This is the problem addressed in this paper.

Standard approaches [4, 5] to assessing semantic content and topic relevance, both for essays and speech, are based on measuring the similarity between vector representations [6, 5] of responses and prompts. These systems need to have seen in training prompt-response pairs for all prompts in a test to assess the relevance of a test response. This limits the flexibility and increases the cost of deployment of such systems, as example responses have to be collected for newly introduced prompts. Re-training the system may also be computationally costly. This limitation is overcome in the approach proposed in [7], called the Attention-based Topic Model (ATM). The ATM allows the assessment of relevance to prompts not seen in the training data without the need to train on associated example responses. This is done by using a bidirectional Recurrent Neural Network (BiRNN) embedding of the prompt to attend over the hidden states of a BiRNN embedding of the response using an attention mechanism to compute a fixed-length embedding which is used to predict relevance. The ATM is trained on a set of matched on-topic (positive) and mismatched off-topic (negative) example prompt-response pairs. Unfortunately, while the system achieves excellent performance on prompts seen in training, performance on unseen prompts is not as good. Furthermore, different prompts may be linked to a broader topic such as business development. This could be used to help predict the relevance of an unseen prompt on a related topic but is not explicitly exploited in the ATM. Also [7] uses only a fixed set of negative example prompt-response matchings during training which lacks diversity.

This paper presents several extensions to the ATM. Firstly, a hierarchical variant of the ATM (HATM) is proposed in an attempt to improve performance on unseen prompts by explicitly exploiting the similarity between prompts via a second attention mechanism, which interpolates all prompts seen in the training data given a prompt of interest. This allows the construction of a prompt ontology and makes the model more interpretable. Furthermore, the entropy of the prompt attention mechanism can be used as a measure of uncertainty of the model, allowing back-off to human assessors. Secondly, a sampling mechanism is added to dynamically generate

This research was funded under the ALTA Institute, Cambridge University. Thanks to Cambridge English Language Assessment for supporting this research and providing access to the BULATS data.

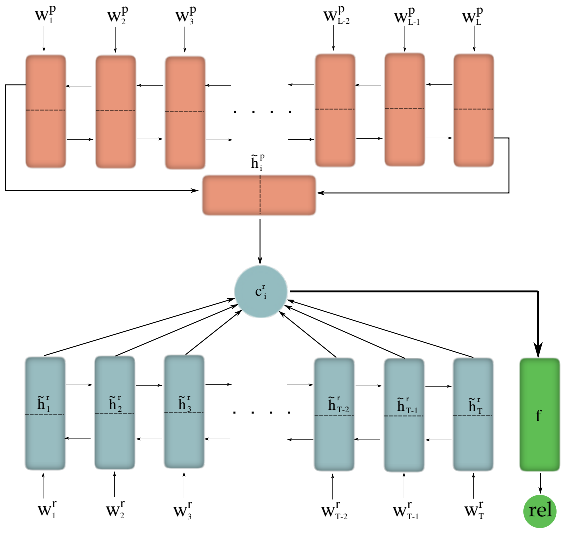


Fig. 1. Attention-based Topic Model

negative examples during training and the use of ASR confidence scores is investigated. Finally, an extensive analysis of the behavior of both models is done. The ability of these models to assess the relevance and detect off-topic responses to prompts which are both seen, and crucially, not seen in the training data is demonstrated on spoken data from the Cambridge Business Language (BULATS) exam.

The rest of this paper is structured as follows: section 2 introduces and describes the proposed models, section 3 describes the data and experimental setup, section 4 contains the experimental results and analysis, and section 5 is the conclusion.

2. MODEL

This section describes the Attention-based Topic Model [7] for assessing the relevance of responses to prompts and highlights its limitations in assessing relevance to unseen prompts. The Hierarchical Attention-based Topic Model is proposed to potentially overcome these limitations by explicitly exploiting the implicit, varying, similarities between prompts.

2.1. Attention-based Topic Model

The ATM [7] (Fig. 1) consists of a prompt encoder (Fig. 1-red), a response encoder and an attention mechanism over responses (Fig. 1-blue) and a binary classifier (Fig. 1-green). The ATM assesses the relevance of responses to prompts by learning to dynamically compute a representation or embedding of the prompt using the prompt encoder. This is used to attend over a representation of the response computed using a response encoder via an attention mechanism. This highlights the parts of the response most relevant to the prompt. Based on this information, a binary classifier assigns the probability of the response being relevant to the prompt.

The prompt (eq. 1) and response (eq. 2) encoders are Bidirectional Recurrent Neural Networks (BiRNN) [8] with LSTM recurrent units [9, 10] which process the word sequences $\boldsymbol{w}^p = \{w_1^p, \cdots, w_L^p\}$ and $\boldsymbol{w}^r = \{w_1^r, \cdots, w_T^r\}$ of the prompt and response, respectively. The prompt embedding $\tilde{\boldsymbol{h}}^p$ is computed by concatenating the final forward in time $\overrightarrow{\boldsymbol{h}}{}^p_L$ and backward in time $\overleftarrow{\boldsymbol{h}}{}^p_1$ hidden states of the prompt encoder (eq. 1). The forward in time $\overrightarrow{\boldsymbol{h}}{}^r_t$ and backward in time $\overleftarrow{\boldsymbol{h}}{}^r_t$ hidden states of the response encoder are concatenated at every time step to produce a hidden state $\tilde{\boldsymbol{h}}^r_t$ (eq. 2), which contains information about how the complete surrounding context relates to the current word.

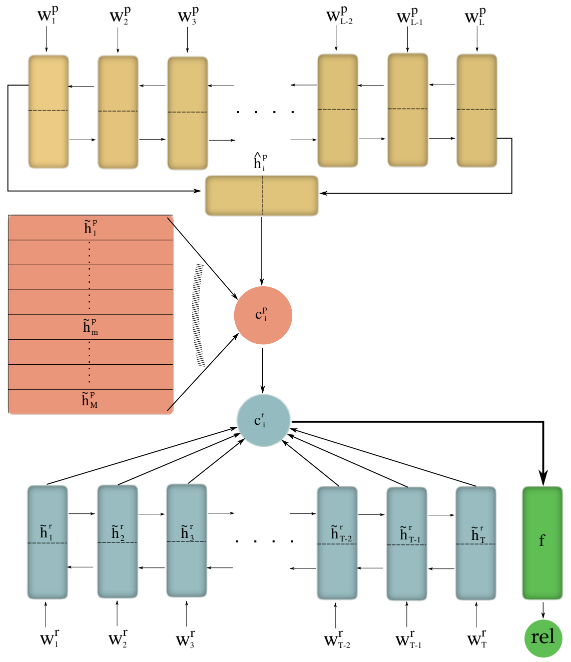


Fig. 2. Hierarchical Attention-based Topic Model

$$\boldsymbol{h}^p_{1:L} = \begin{bmatrix}\overrightarrow{\boldsymbol{h}}{}^p_1\\ \overleftarrow{\boldsymbol{h}}{}^p_1\end{bmatrix}\cdots\begin{bmatrix}\overrightarrow{\boldsymbol{h}}{}^p_L\\ \overleftarrow{\boldsymbol{h}}{}^p_L\end{bmatrix} = \texttt{BiLSTM}^p(\boldsymbol{w}^p;\ \boldsymbol{\theta}^p);\ \tilde{\boldsymbol{h}}^p = \begin{bmatrix}\overrightarrow{\boldsymbol{h}}{}^p_L\\ \overleftarrow{\boldsymbol{h}}{}^p_1\end{bmatrix} \tag{1}$$

$$\tilde{\boldsymbol{h}}^r_{1:T} = \begin{bmatrix}\overrightarrow{\boldsymbol{h}}{}^r_1\\ \overleftarrow{\boldsymbol{h}}{}^r_1\end{bmatrix}\cdots\begin{bmatrix}\overrightarrow{\boldsymbol{h}}{}^r_T\\ \overleftarrow{\boldsymbol{h}}{}^r_T\end{bmatrix} = \texttt{BiLSTM}^r(\boldsymbol{w}^r;\ \boldsymbol{\theta}^r) \tag{2}$$

The prompt and response encoders are combined using a fixed-length prompt-conditional embedding $\boldsymbol{c}^r_i$ of the response. This is computed as a weighted sum of the hidden states $\tilde{\boldsymbol{h}}^r_t$ of the response encoder given a set of attention weights α_t via an attention mechanism (eq. 3). The attention weights for each hidden state are computed as a softmax (eq. 4), where the logits are given by a similarity function (eq. 5) which computes how strongly a hidden state of the response encoder relates to the embedding of the prompt. The parameters of the attention mechanism are $\boldsymbol{\theta}^a = \{\boldsymbol{v}_r, \boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \boldsymbol{b}\}$.

$$\boldsymbol{c}^r_i = \sum_{t=1}^{T} \alpha_{i,t}\tilde{\boldsymbol{h}}^r_t \tag{3}$$

$$\alpha_{i,t} = \frac{\exp(s(\tilde{\boldsymbol{h}}^p_i, \tilde{\boldsymbol{h}}^r_t))}{\sum_{\tau=1}^{T}\exp(s(\tilde{\boldsymbol{h}}^p_i, \tilde{\boldsymbol{h}}^r_\tau))} \tag{4}$$

$$s(\tilde{\boldsymbol{h}}^p_i, \tilde{\boldsymbol{h}}^r_t) = \boldsymbol{v}_{\boldsymbol{r}}^{\mathrm{T}} \tanh(\boldsymbol{\Lambda}_1\tilde{\boldsymbol{h}}^p_i + \boldsymbol{\Lambda}_2\tilde{\boldsymbol{h}}^r_t + \boldsymbol{b}) \tag{5}$$

The embedding $\boldsymbol{c}^r_i$ is then fed into a binary classifier $f(\boldsymbol{c}^r_i;\ \boldsymbol{\theta}^f)$ (eq. 6) which generates the relevance probability $\texttt{P}(\texttt{rel}|\boldsymbol{w}^r, \boldsymbol{w}^p)$ of the response relating to the question. In this work f is a deep neural network (DNN) with parameters $\boldsymbol{\theta}^f$.

$$\texttt{P}(\texttt{rel}|\boldsymbol{w}^r, \boldsymbol{w}^p) = f(\boldsymbol{c}^r_i;\ \boldsymbol{\theta}^f) \tag{6}$$

The ATM is trained using minibatch stochastic gradient descent with a logistic loss error function (eq. 7) over all parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}^p, \boldsymbol{\theta}^r, \boldsymbol{\theta}^a, \boldsymbol{\theta}^f\}$

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{N}\sum_{i=1}^{N} & t_i \log(\mathtt{P}(\mathtt{rel}|\boldsymbol{w}_i^r, \boldsymbol{w}_i^p)) \\ & + (1-t_i)\log(1-\mathtt{P}(\mathtt{rel}|\boldsymbol{w}_i^r, \boldsymbol{w}_i^p))\end{aligned} \tag{7}$$

where $t_i = 1$ for positive examples and $t_i = 0$ for negative examples.

2.2. Hierarchical Attention-based Topic Model

The ATM directly embeds a prompt via the prompt encoder, which may fail to generalize well to unseen prompts. The Hierarchical Attention-based Topic Model (HATM) (Fig. 2) extends the ATM to explicitly make use of the similarity between prompts. For example, in a business test prompts might relate to different aspects of a company's business or an individual's personal development or organizing events etc. The HATM assumes that there is an implicit ontology of prompts in the data, which the HATM learns in an unsupervised fashion. Prompts are expressed as a weighted sum of prompts $\tilde{\boldsymbol{h}}^p$ seen in the training data (Fig. 2-red), where the weights, given by a prompt attention mechanism (eq. 10, 11) (Fig. 2-red), have a sum-to-one constraint. This yields a new prompt embedding $\boldsymbol{c}^p$ (eq. 9) which is used to attend over the responses. Thus, the HATM views seen prompts as the vertices of a simplex within which all valid prompt embeddings must exist. This potentially allows the HATM, given a robust and diverse set of prompt embeddings, to estimate embeddings for unseen prompts more robustly. Furthermore, the learned ontology may be useful for determining which prompts are more and which are less confusable. The prompts seen in the training data never directly attend over themselves - the attention mechanism is trained in a 'leave-one-out' fashion to teach it to reconstruct each prompt in the training data from all other seen prompts. The prompt attention mechanism uses a separate 'search' embedding (eq. 8) given by a search encoder (Fig. 2-yellow) of the prompt $\hat{\boldsymbol{h}}^p$ to compute attention over the prompt embeddings $\tilde{\boldsymbol{h}}^p$. The parameters of the prompt attention mechanism are $\boldsymbol{\theta}^{pa} = \{\boldsymbol{v}_p, \boldsymbol{\Lambda}_1^p, \boldsymbol{\Lambda}_2^p, \boldsymbol{b}^p\}$, thus two new sets of parameters are added to the system: $\{\boldsymbol{\theta}^{pa}, \boldsymbol{\theta}^s\}$.

$$\hat{\boldsymbol{h}}^p = \begin{bmatrix}\overrightarrow{\boldsymbol{h}}^s_L \\ \overleftarrow{\boldsymbol{h}}^s_1\end{bmatrix};\ \boldsymbol{h}^s_{1:L} = \mathtt{BiLSTM}^s(\boldsymbol{w}^p;\, \boldsymbol{\theta}^s) \tag{8}$$

$$\boldsymbol{c}_i^p = \sum_{m=1}^{M} \alpha^p_{i,m}\tilde{\boldsymbol{h}}^p_m \tag{9}$$

$$\alpha_{i,m} = \begin{cases}\dfrac{\exp(s(\hat{\boldsymbol{h}}_i^p, \tilde{\boldsymbol{h}}_m^p))}{\sum_{m=1,\neq i}^{M}\exp(s(\hat{\boldsymbol{h}}_i^p, \tilde{\boldsymbol{h}}_m^p))}, & \text{if } i \neq m \\ 0, & \text{if } i = m\end{cases} \tag{10}$$

$$s(\hat{\boldsymbol{h}}_i^p, \tilde{\boldsymbol{h}}_m^p) = \boldsymbol{v}_{\boldsymbol{p}}^{\mathrm{T}}\tanh(\boldsymbol{\Lambda}_1^p\hat{\boldsymbol{h}}_i^p + \boldsymbol{\Lambda}_2^p\tilde{\boldsymbol{h}}_m^p + \boldsymbol{b}^p) \tag{11}$$

3. DATA AND EXPERIMENTAL SETUP

A series of experiments were run to evaluate the ability of the ATM and HATM to assess the relevance of responses to prompts. Data from the Business Language Testing Service (BULATS) English tests was used for training and test. The text for each response was generated using an ASR system. The 1-best recognition hypothesis was then passed to a relevance assessment system (ATM/HATM), which decided whether the candidate had spoken off topic by assigning a probability of whether the response was relevant to the prompt. To avoid a data mismatch, the recognition hypotheses were used both in training and test.

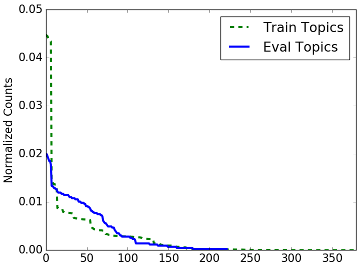


Fig. 3. Topic distributions in order of decreasing topic count

3.1. BULATS Test Format and Data

The BULATS Online Speaking Test has five sections [11]. This work focuses on the 3 sections where open ended prompts (which appear on screen) elicit spontaneously constructed responses. In Section C, candidates talk about a work related topic (e.g. the perfect office). Candidates must describe a graph such as a pie or bar chart related to a business situation (e.g. company sales) in Section D. In Section E candidates are asked to respond to 5 prompts related to a single context prompt (e.g. a set of 5 questions about organizing a stall at a trade fair). There are 7 prompts in total.

Data	#Topics	#Resp.	#Words	#Resp./	Avg.Resp.
				Topic	Length
TRN	379	293.0K	13.4M	773.2	45.8
EVAL1	92	1297	64.4K	14.1	49.7
EVAL2	177	1335	58.5K	7.5	43.8
EVAL3	179	1445	63.1K	8.1	43.7
ALL	219	4077	186.0K	18.6	45.6

Table 1. Topic, response and word statistics of the prompt-response data sets based on 1-best recognition hypotheses.

Table 1 gives the statistics of the prompt-response data sets used in this paper. Each prompt corresponds to one topic, making the terms interchangeable. The training data set *TRN* contains 13.4M words in 293.0K responses from 42K candidates. 379 unique prompts are seen in *TRN*, with an approximately Zipfian distribution (Fig. 3). There are an average of 773 example responses per topic (prompt), with an average response length of 45.8 words. *TRN* has a wide range of candidate L1s, with the largest proportion being Gujarati L1. The evaluation data sets, *EVAL1-3, ALL*, are designed to have an (approximately) even distribution over CEFR grades levels [12]. Fig. 3 shows that the topic distribution of the evaluation data is less skewed than the training data but still roughly Zipfian. *EVAL1* is composed of only Gujarati L1 speakers, *EVAL2* of only Spanish L1 candidates and *EVAL3* is composed of Arabic, Dutch, French, Polish, Thai and Vietnamese L1 candidates. The evaluation data set *ALL* is the combination of *EVAL1-3*. A subset of inappropriate responses, where the speaker failed to provide a meaningful response, was removed from the evaluation data, which may cause some minor differences with the results quoted in previous work [7].

3.2. Training Data Construction

As the data is taken from tests run with human examiners the responses are assumed to be on topic, with the exception of the previously mentioned subset of inappropriate responses which were removed. To produce negative, off-topic training examples the responses and prompts were shuffled during training via a dynamic sampling mechanism which samples mismatched prompts to a given response. This is in contrast to [7], where a fixed set of negative examples was created and maintained during training. The sampling mechanism can draw topics from the empirical topic distribution (Fig. 3) or from a uniform distribution. Positive examples come from the empirical topic distribution. If more than one negative example is shown for a particular response, the positive example is over-sampled the corresponding number of times to maintain a balanced training. As was shown in [13, 7], prompts from the same section tend to be more similar, and therefore more confusable. Two topic shuffling strategies were considered in [13, 7]: *Naive*, where prompts are shuffled across all sections; and *Directed*, where prompts are shuffled only within the same section. This work only considers *Naive* topic shuffling, as it represents the more likely scenario - real off-topic responses are unlikely to come predominantly from the same section. For multi-part prompts, which contain a main prompt that describes the overall question, and several (5 here) sub-prompts, all sub-prompts were pre-appended with the main prompt. These sub-prompts are considered distinct topics and thus competing negative examples to each other during shuffling.

3.3. ASR System

In this work a speaker independent hybrid DNN-HMM ASR system [14] is used. The acoustic model is trained on 108.6 hours of BULATS test data (Gujarati L1 speakers) using the HTK v3.5 toolkit [15, 16]. A Kneser-Ney trigram language model is trained on this data and interpolated with a general English language model trained on a large broadcast news corpus, using the SRILM toolkit [17]. The performance of this ASR system is described in Tables 2 and 3 relative to crowd-sourced transcriptions [18]. The crowd-sourced transcriptions are more accurate but mismatched to the ASR transcriptions used in the ATM training. Very little difference was observed between the performance of the ATM on ASR and crowd-sourced transcriptions of the evaluation data. Results on crowd-sourced transcriptions are thus not quoted for brevity.

EVAL1	EVAL2	EVAL3	ALL
37.3	52.5	48.6	45.7

Table 2. ASR %WER on evaluation data sets

A1	A2	B1	B2	C
60.3	54.0	44.9	41.8	41.4

Table 3. ASR %WER per CEFR grade level on *ALL*

3.4. Model and Training Hyper-parameters

The ATM and HATM were implemented in Tensorflow [19] and contain two 400 dimensional BiLSTM encoders with TanH nonlinearities, 200 for the forward states and 200 for the backward states. The HATM also contains an additional 200-dimensional BiLSTM prompt-search encoder. The ATM was trained for 5 epochs with the Adam optimizer [20], an exponentially decaying learning rate with an initial value of 1e-3 and decay factor 0.85 per epoch. Dropout regularization [21] was applied to all layers except for the LSTM recurrent connections and word embeddings, with a keep probability of 0.8. The binary classifier was a DNN with 2 hidden layers of 200 rectified linear (ReLU) units and a 1-dimensional logistic output. The word embeddings, shared by all BiLSTMs, were initialized from an RNNLM language model trained on the *TRN* responses and kept fixed during training. The HATM was initialized from a trained ATM. For the first 3 epochs only the newly-initialized prompt-attention mechanism was trained. Further training for 1 more epoch is done with an unlocked response attention mechanism and a learning rate of 1e-4. The prompt and response encoders, as well as the DNN classifier remain locked. The ATM takes about 3.3 hours on an nVidia GTX 980M graphics card. Further training the HATM takes an extra hour.

3.5. Assessment Criteria

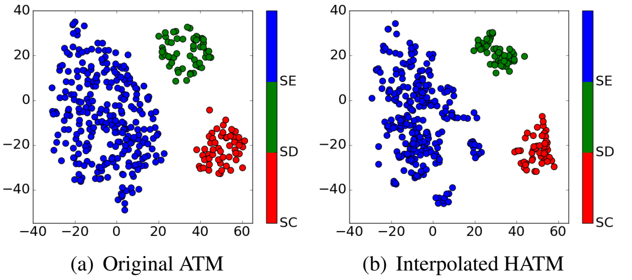
The models are evaluated using the area under a Receiver-Operator Characteristic (AUC), which plots the True Positive vs. the False Positive rate at different decision thresholds. To yield this, negative examples (true negatives) need to be introduced into the evaluation data sets via shuffling. The negative examples are drawn from the empirical topic distribution of the evaluation data. It must be noted that results are based on a particular instance of shuffling the prompts for evaluation.

4. EXPERIMENTAL RESULTS

This section presents the results of investigations into the properties of the ATM and HATM. Subsection 4.1 investigates several key properties of the models when all the prompts are seen. Firstly, the effect of dynamically sampling negative examples from the empirical and uniform distributions is assessed and compared to fixed sampling. Performance is compared to the ATM model trained in [7]. Secondly, the performance of the ATM and HATM is compared. Finally, the nature of the prompt attention mechanism in the HATM is investigated. Subsection 4.2 investigates the performance of the ATM and HATM on unseen topics (prompts), analyses errors which the models make and compares results to previous work. Lastly, subsection 4.3 investigates whether incorporating confidence scores from the recognition hypotheses in the model input features can improve performance.

4.1. Baseline Performance

Table 4 shows the baseline performance of the ATM model using fixed, empirical and uniform sampling of prompts for negative training examples. Results quoted for fixed sampling are made using the same models which were used in previous work [7]. Empirical sampling yields better performance than fixed sampling when using 1 negative example per response. This is because dynamically sampling topics is an efficient way of increasing the diversity of negative examples. Uniform sampling yields the worst performance. This is probably because the mismatch in the topic distributions of the positive and negative training examples skews the model towards treating rare topics as not relevant. Additionally, there is a mismatch to the topic distribution of the evaluation data. Thus, models with uniform sampling were not further investigated. Using 5 negative examples per response produced an improvement in the AUC only for fixed negative sampling, yielding the highest overall AUC of 0.98. This is because the diversity of negative examples is increased, as for the



(a) Original ATM (b) Interpolated HATM

Fig. 4. t-SNE projection of prompt embeddings

empirical sampling, but also all rare topics in the data are definitely seen with fixed sampling. Though dynamic empirical sampling is statistically equivalent to fixed sampling, rare topics may in practice not be handled well given a limited number of epochs.

#neg. exs.	Fixed	Empirical	Uniform
1	0.96	0.97	0.95
5	0.98	0.97	0.95

Table 4. Comparison of AUC for ATM models with fixed, empirical and uniform negative sampling on *ALL*

ATM	HATM
0.97	0.96

Table 5. Comparison of ATM and HATM performance on *ALL*

The performance of the ATM and HATM is compared in table 5. It can be seen that the HATM performs marginally worse than the ATM. It is interesting to investigate what the prompt attention mechanism and the prompt encoder have learned in the HATM. Firstly, a t-SNE [22] projection of the original (ATM) prompt embeddings (Fig. 4a) is compared to the projection of the interpolated HATM embeddings (Fig. 4b). Both sets of embeddings form three distinct clusters, grouped by section. Notably, the interpolated embeddings reside in the same locations as the originals, though they appear to be more tightly grouped. The attention mechanism also learns section distinctions well and the confusion matrix (not shown) between prompt sections shows that the the HATM predominantly attends only over prompts of the corresponding section.

Fig. 5 shows the entropy of the attention mechanism for all topics, sorted in order of decreasing entropy. AUC performance of the HATM was assessed on subsets of *ALL* which contain an increasing fraction of topics, added in order of decreasing entropy. The plot of AUC on these subsets is also shown on Fig. 5. AUC increases as lower entropy topics are added to the data. This shows that there are topics which the HATM is able to understand and focus on well, and others for which it struggles to confidently find a similar topic. This suggests that the entropy of the prompt attention mechanism can be seen as a measure of uncertainty of the HATM's ability to accurately assess relevance. Thus, depending on the prompt topic, all responses to a given prompt whose entropy is above a threshold could be rejected by the model and e.g. passed to be assessed by a human, with other responses processed automatically. Furthermore, if new, unseen prompts are added, the entropy can be used as an indicator of whether the model can perform adequately on them, or if example responses to the new prompt need to be collected and the model retrained. This is an important advantage of the HATM over the ATM, despite their comparable performance.

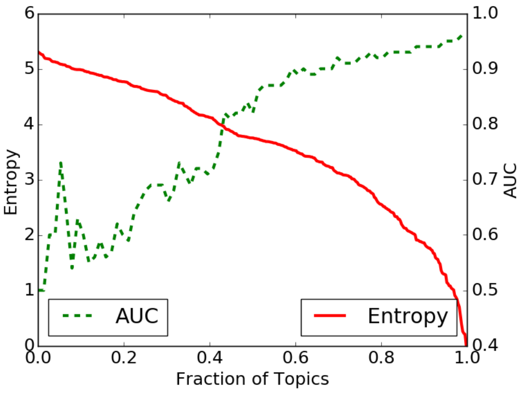


Fig. 5. HATM prompt attention

4.2. Performance on Unseen Prompts

The proposed models' ability to generalize to new prompts is investigated in this section. Since real unseen prompt-response pairs are unavailable, 10-fold cross validation over prompts (topics) was used on the training and evaluation data. A fixed block of data, *TRN-fixed* (Table 6), is never removed from the training data, as it contains topics which dominate the training data and topics which do not appear in the evaluation set *ALL*. The *TRN-xVal* data was used in cross validation. A subset of *ALL*, called *ALL-sub*, without the dominant topics of *TRN*, was used for cross validation evaluation. All parts of related multi-part prompts are held out together.

Data	#Topics	#Resp.	#Words
TRN-fixed	178	142.8K	6.8M
TRN-xVal	201	150.1K	6.6M
ALL-sub	201	2955	127.7K

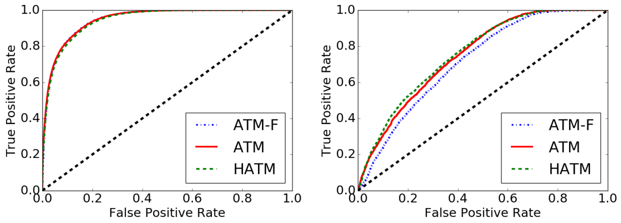
Table 6. Topic, response and word statistics of the prompt-response data sets used for 10-fold cross validation

The evaluation prompts presented to the models in the following experiments are always either from subsets which are seen or unseen in the training data. As in section 4.1, evaluation responses are always new (not reused from the training data), but can be related to prompts either seen or unseen in training. Two strategies for shuffling evaluation responses for negative examples are considered: *seen*, *unseen*. The first uses responses to seen prompts as negative examples, the second uses responses to unseen prompts as negative examples. This produces four experiments which illustrate different aspects of how well the models understand what relates to seen prompts and how well they generalize to new, unseen prompts. Relevance probabilities are combined across all 10 folds to produce one ROC curve and AUC score for each experiment. These curves, and the associated AUC scores, represent the 'average' AUC on the data. To decrease noise arising from particular shufflings of the evaluation data, 10 different random topic shufflings are used as negative examples and the positive examples are replicated 10 times for all 10 cross-validations folds.

The results in Table 7 lines 1-2 show that once prompts have been seen in training, the model has a clear understanding of what

Prompts	System	Neg. Resp. relating to	
		Seen Prompts	Unseen Prompts
Seen	ATM	0.949	0.938
	HATM	0.944	0.933
Unseen	ATM	0.855	0.751
	HATM	0.856	0.760

Table 7. Average AUC on *ALL-sub*



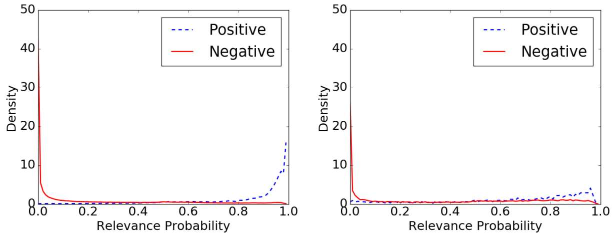
(a) Seen Seen (b) Unseen Unseen

Fig. 6. ROC curves

is relevant to them and is generally not sensitive to the nature of the negative-example responses. However, on unseen prompts (Table 7 lines 3-4) there is a degradation of performance, which ranges from 0.855 down to 0.751 for the ATM and from 0.856 to 0.760 for the HATM as the topic shuffling of the negative responses changes from *seen* to *unseen*. Clearly, the models are able to generalize well to and assess the relevance of unfamiliar responses to seen prompts, and to a lesser degree, are able to perform reasonably on new and unseen prompts, even in the extreme scenario (0.760 AUC). This is expected, as the models are exposed to a greater variety of responses than prompts. ROC curves for the performance on seen and unseen prompts for positive responses with corresponding negative response topic shuffling are shown in Figs. 6a, 6b. The ROC curves demonstrate a small but consistent gain (0.760 vs 0.751) of the HATM over the ATM on unseen prompts. Both the ATM and HATM consistently outperform the fixed sampling ATM model (ATM-F) from [7], which has a AUC of 0.717 on unseen prompts with unseen responses.

It is interesting to analyze the mistakes which the system makes. To do this, the relevance probabilities for positive and negative examples are plotted as histograms for the scenarios where seen prompts are used to generate both sets of examples (Fig. 7a) and where unseen prompts are used for both (Fig. 7b). The other scenarios yield similar histograms. When operating on seen prompts, the model is able to correctly classify most examples with very high/low relevance probabilities. However, when operating on unseen prompts it is able to confidently detect when prompts and responses are mismatched, but is unsure about matched prompt-response pairs for unseen prompts. This is the main failure case of these models.

This suggests that the models, via the response attention mechanism, learn a 'lock and key' mechanism, where for a given response, only summation of the hidden states using weights derived from a matched prompt result in a high relevance prediction, and all other summations in a low relevance prediction. In the matched case for unseen prompts the models correctly do not yield a very low relevance score, but struggle to yield a high relevance score, which indicates a generalization issue. It should be noted that 'lock and key' behavior reflects the way the models are trained - each response in the training data is used as a positive example only once, when matched with an appropriate prompt, and many times as a negative example, when matched with any other prompt.



(a) Seen Seen (b) Unseen Unseen

Fig. 7. Relevance probability histograms

4.3. Use of ASR confidence Scores

The ASR system can output confidence scores for each word which give a measure of how likely the recognized word is correct. These scores could potentially help the ATM by focusing attention on words which the recognizer is more confident about. Word level confidence scores from the ASR output were therefore applied to modify the ATM input. The mapped (to remove biases [23]) confidence scores were used as an extra input into the response attention mechanism (ATM-C). The response similarity function was modified to use the confidence score γ_t^r as an extra scaled bias:

$$s(\tilde{\boldsymbol{h}}_i^p, \tilde{\boldsymbol{h}}_t^r, \gamma_t^r) = \boldsymbol{v}_{\boldsymbol{r}}^{\mathrm{T}} \tanh(\boldsymbol{\Lambda}_1 \tilde{\boldsymbol{h}}_i^p + \boldsymbol{\Lambda}_2 \tilde{\boldsymbol{h}}_t^r + \boldsymbol{b}_\gamma \gamma_t^r + \boldsymbol{b}) \qquad (12)$$

From Table 8 it can be seen that currently no benefit was observed from using confidence scores.

ATM	ATM-C
0.97	0.97

Table 8. Effect on AUC of using ASR confidence scores in the ATM

5. CONCLUSIONS AND FUTURE WORK

This paper has addressed the detection of off-topic responses for automatic spoken language assessment, in particular to prompts unseen in training using the Attention-based Topic Model (ATM) [7]. The ATM uses negative off-topic prompt-response examples in training. Switching from fixed to dynamic sampling of these examples gave a consistent performance gain in detecting the relevance of unseen prompts to unseen responses. The Hierarchical Attention-based Topic Model (HATM) was proposed to extend the ATM to explicitly make use of the relationships between prompts through a prompt attention mechanism. The ATM and HATM perform similarly, with a small but consistent gain with the HATM on unseen prompts. The key advantage of the HATM is that it returns a measure of uncertainty in its ability to assess relevance via the prompt attention mechanism, which allows it to back-off to human assessors, increasing robustness. The behavior and primary failure modes of the ATM and HATM were analyzed, and it was determined that the models primarily fail to assign a high relevance probability to unseen prompts with matched relevant unseen responses. Future work should investigate data augmentation strategies to deal with the heavily skewed topic distribution of the training data, the use of higher quality ASR transcriptions, and further investigate the use of ASR confidence scores.

6. REFERENCES

[1] Barbara Seidlhofer, “English as a lingua franca,” *ELT journal*, vol. 59, no. 4, pp. 339, 2005.

[2] Klaus Zechner et al., “Automatic scoring of non-native spontaneous speech in tests of spoken English,” *Speech Communication*, vol. 51, no. 10, pp. 883–895, 2009.

[3] Angeliki Metallinou and Jian Cheng, “Using Deep Neural Networks to Improve Proficiency Assessment for Children English Language Learners,” in *Proc. INTERSPEECH*, 2014.

[4] Helen Yannakoudakis, “Automated assessment of English-learner writing,” Tech. Rep. UCAM-CL-TR-842, University of Cambridge Computer Laboratory, 2013.

[5] Thomas K Landauer, Peter W. Foltz, and Darrell Laham, “Introduction to Latent Semantic Analysis,” *Discourse Processes*, vol. 25, pp. 259–284, 1998.

[6] Xuan-Hieu Phan and Cam-Tu Nguyen, “GibbsLDA++: A C/C++ implementation of latent Dirichlet allocation (LDA),” `http://gibbslda.sourceforge.net/`, 2007.

[7] Malinin A, K. Knill, A. Ragni, Y. Wang, and M.J.F. Gales, “An attention based model for off-topic spontaneous spoken respnse detection: An Initial Study,” in *to be presented at ISCA Workshop on Speech and Language Technology for Education (SLaTE)*, 2017.

[8] Mike Schuster and Kuldip K Paliwal, “Bidirectional recurrent neural networks,” *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[9] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[10] Alex Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, Studies in Computational Intelligence, Springer, 2012.

[11] Lucy Chambers and Kate Ingham, “The BULATS Online Speaking Test,” *Research Notes*, vol. 43, pp. 21–25, 2011.

[12] Council of Europe, *Common European framework of reference for languages: Learning, teaching, assessment*, Cambridge, U.K: Press Syndicate of the University of Cambridge, 2001.

[13] Andrey Malinin, Rogier van Dalen, Kate Knill, Yu Wang, and Mark Gales, “Off-topic Response Detection for Spontaneous Spoken English Assessment,” in *Proc. 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, Berlin, Germany, 2016, pp. 1075–1084.

[14] Haipeng Wang et al., “Joint Decoding of Tandem and Hybrid Systems for Improved Keyword Spotting on Low Resource Languages,” in *Proc. INTERSPEECH*, 2015.

[15] Steve Young et al., *The* HTK *book (for* HTK *Version 3.4.1)*, University of Cambridge, 2009.

[16] Steve Young et al., *The* HTK *book (for* HTK *version 3.5)*, University of Cambridge, 2015, `http://htk.eng.cam.ac.uk`.

[17] A. Stolcke, “SRILM an extensible language modelling toolkit,” in *Proc. International Conference on Spoken Language Processing (ICSLP)*, 2002.

[18] Rogier C. van Dalen, Kate M. Knill, Pirros Tsiakoulis, and Mark J. F. Gales, “Improving Multiple-Crowd-Sourced Transcriptions Using a Speech Recogniser,” in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2015.

[19] Martín Abadi et al., “TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems,” 2015, Software available from tensorflow.org.

[20] Diederik P. Kingma and Jimmy Ba, “Adam: A Method for Stochastic Optimization,” in *Proc. 3rd International Conference on Learning Representations (ICLR)*, 2015.

[21] Nitish Srivastava et al., “Dropout: a simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[22] L. van der Maaten and G. Hinton, “Visualizing Data using t-SNE,” *J. MLR*, vol. 1, pp. 1–49, 2008.

[23] G. Evermann and P. C. Woodland, “Large vocabulary decoding and confidence estimation using word posterior probabilities,” in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2000.