Salience-Aware Temporal Attention-Gated Model for Robust Sequence Classification

Anonymous CVPR submission

Paper ID ****

Abstract

Traditional techniques for sequence classification are typically designed for the well-segmented sequence data which has been edited to remove noisy or irrelevant parts. Therefore, such methods cannot be easily applied on noisy sequences which can be expected in real-world applications. We assimilate ideas from the attention model and gated recurrent networks to build a new sequence classification model which is able to deal with the raw sequence with noise. Specifically, we employ an attention model to measure the relevance of each time step of a sequence to the final decision. We then use the relevant segments based on their attention scores in a novel gated recurrent network to learn the hidden representation for the classification. More importantly, Our attention weights provide a physically meaningful interpretation for the salience along the time domain. We demonstrate the potential merits of our model named Temporal Attention-Gated Model (TAGM) in both interpretability and classification performance of our model on a variety of tasks, including speech recognition, textual sentimental analysis and event recognition.

1. Introduction

Sequence classification is posed as a problem of assigning a single label to a sequence of observations. The sequence classification models can be applied to extensive applications ranging from computer vision to natural language processing. Most existing sequence classification models (e.g., hidden-state conditional random field [29] and hidden-unit logisitic model [25]) follow the basic framework that focuses on learning an effective hidden representation in a supervised way to capture both the latent structure in feature space and temporal information along the time domain. These types of methods are designed for the well segmented sequences without irrelevant and noisy parts that could mislead the classifier. As a result, these models require the input sequences to be pre-processed to remove the irrelevant subsequences, thereby avoiding the interference by the irrelevant information. However, the pre-processing step is normally performed in a handcrafted way and hence quite inconvenient and expensive for various real-world tasks which do not provide pre-segmented data. The problem can be circumvented by gated recurrent networks like gated recurrent units [4] and long short-term memory [11]. They employ gates to balance the information flow from current and previous time steps. Nevertheless, these methods model the gates w.r.t. each hidden unit instead of whole time step, thus it is hard to to interpret the importance of each time step for the final decision. Alternatively, another way to detect salient segments is the adoption of attention-based mechanism, which models how much attention should be paid to a specific segment.

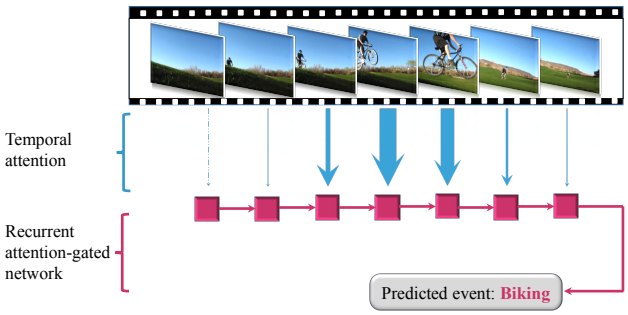

Figure 1. The model proposed in this work first employs an attention module to extract the salient frames from the raw noisy input sequences, and then learns an effective hidden representation for the top classifier. The wider the flow line is, the more the information is incorporated into the hidden representation. The dash line represents the zero input.

In this work, we combine the ideas from attention models and gated recurrent networks to propose an attention-based sequence classification model which is able to automatically localize the salient segments which are relevant to the final decision and ignore the irrelavant (noisy) parts given a raw sequence. As a consequence, the decision made based on the selected relevant segments is more accu-

rate than the conventional models that take into account the whole sequence. We refer to the resulting model as Temporal Attention-Gated Model (TAGM). Figure 2 presents an intuitive overview for it.

Notably, compare to conventional sequence classification models, the proposed approach benefits from following potential advantages:

- It is able to automatically capture the salient parts of the input sequences thereby leading to better performance.
- The inferred salience scores provide a physically meaningful interpretation with respect to the informativeness of the raw input sequence along the time domain.
- Compared to conventional gated recurrent models such as LSTM, our model reduces the number of parameters which leads to faster training and inference and better generalizability with less training data.

2. Related Work

This work involves mainly in three research subareas including sequence classification, attention models and recurrent networks. Each of them owns a substantial amount of prior work which we cannot fully cover due to the limited space.

Sequence Classification. Most of existing sequence classification models are specifically designed for the well-segmented sequence without noise contained inside. They can be divided roughly into two categories:

1. The models focusing on learning an effective intermediate representation based on generative models for the subsequent standard classifiers such as SVMs. These methods are typically based on the use of (kernels based on) the hidden Markov models (HMMs) [30] or dynamic time warping (DTW) [15]. The HMM is a generative model which models the sequence data in a chain of latent k-nomial features. It can be extended to class-conditional HMMs for sequence classification by combining class priors via Bayes' rules. HMM can also be used as the base model for Fisher Kernel [14] to learn a sequence representations.

2. Discriminative graphical models which model the distribution over all class labels conditioned on the input data. Conditional random fields (CRF) [21] are discriminative models for sequence labeling. A potential drawback of common linear-chain CRFs is that the linear nature cannot model complex decision boundaries. To address this limitation, many models (e.g., latent-dynamic CRFs [28], conditional neural fields [26], neural conditional random fields [5] and hidden-unit CRF model [37]) are proposed to model the latent nonlinear structure hidden in the data. Hidden-state CRF (HCRF) [29] employs a chain of k-nomial latent variables to model the latent structure and has be successfully used in the sequence classification. Similarly, hidden unit logistic model (HULM) [25] utilizes binary stochastic hidden units to represent the exponential hidden states so as to model more complex latent decision boundaries.

Aforementioned priors works all aim at the well-segmented sequence classification, which cannot cope with the noisy sequence. The model proposed in this work attempts to address this limitation by first filtering out the noise.

Attention Models. Inspired by the attention scheme of human foveal vision, attention model is proposed to focus selectively on certain relevant parts of the input by measuring the sensitivity of output to variances of the input. Doing so can not only improve the performance of the model but also result in the better interpretability. Attention models have been applied to image or video captioning [39, 3, 6, 40], machine translation [1, 23, 31], depth-based person identification [10], speech recognition [8]. Our model employs the attention model to measure the relevance of each time step and then serves as the gate value for the recurrent hidden representation learning.

Recurrent Networks. Recurrent Neural Networks (RNN) learn a representation for each time step taking into account both the observation at current time step and the representation in the previous time step [32]. The biggest advantage of recurrent neural networks lies in the capability of preserving information over time by the recurrent mechanism of the models. Recurrent networks have been successfully applied to various tasks such as language [24], image generation [36], online handwriting [7] and so on. To address the gradient vanishing problem of plain-RNN when dealing with long sequence, LSTM [11] and GRU [4] are equipped with the gates to balance the information flow from the previous time step and current time step dynamically. Specifically, they learn the gate value for each hidden unit along with the recurrent iterations proceed. Inspired by this setup, our model also employs a gate to filter out the noise time steps and reserve the salient ones. But the difference from the LSTM and GRU is that the gate value in our model is fed from the attention module which focuses on the learning the salience of each time step.

3. Temporal Attention-Gated Model

Suppose we are given a noisy sequence (e.g., a video) in which only some frames are useful for the final task (e.g.,

event recognition). Conditioned on this input sequence, the proposed model named Temporal Attention-Gated Model is able to (1) calculate an salience mechanism over the input sequence to measure relevant segments (salience) contributing to the final decision, (2) learn the hidden representation based on the attention score and perform the final supervision task. The model can be trained in an end-to-end manner efficiently. The graphical structure of the model is illustrated in Figure 2.

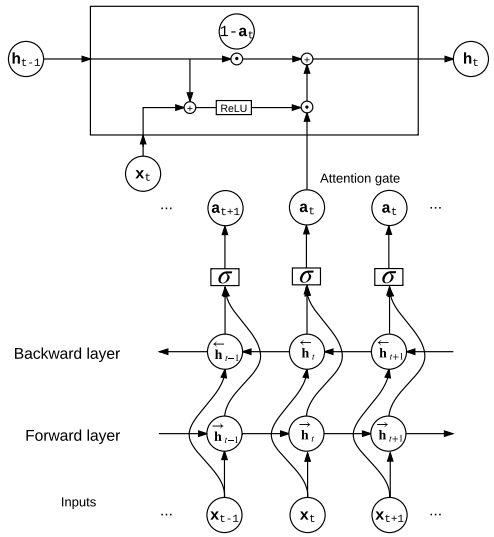

Figure 2. The graphical representation of the Recurrent Attention-Gated Model. Note that a_t is a scalar value instead of a vector, hence $\odot$ in the figure means multiplication between a scalar and a vector.

3.1. Recurrent Attention-Gated Networks

Due to the capability of capturing the temporal information contained in the sequence, Recurrent Neural Network (RNN) is employed to learn the hidden representation for the input sequence. In order to extract the relevant salience sections and ignore the irrelevant parts, we define an attention gate inserted into the recurrent unit in each time step to control how much information is incorporated from the input of the current time step based on the relevance to the final supervised task.

Specifically, given a raw sequence $\mathbf{x}_{1,\ldots,T} = \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$ of length T in which $\mathbf{x}_t \in \mathbb{R}^D$ denotes the observation at the t-th time step, the attention gate at time step t is denoted as a_t, which is a scalar value that indicates the relevance of current time step to the final decision. The hidden state at time step t is modeled as:

$$\mathbf{h}_t = (1 - a_t) * \mathbf{h}_{t-1} + a_t * \mathbf{h'}_t \tag{1}$$

Wherein, $\mathbf{h'}_t$ is the candidate state value which fully incorporates the input information $\mathbf{x}_t$ in the current time step:

$$\mathbf{h'}_t = g(\mathbf{W} \cdot \mathbf{h}_{t-1} + \mathbf{U} \cdot \mathbf{x}_t + \mathbf{b}) \tag{2}$$

Herein, $\mathbf{W}$, $\mathbf{U}$ are respectively the linear transformation parameters for previous and current time steps while $\mathbf{b}$ is the bias term. The model would make a balance between current candidate hidden state and previous hidden state with attention gate a_t. High attention value would push the model to focus more on the current hidden state and input feature, while low attention value would make the model ignore the current input feature and inherit more information form previous time steps.

The learned hidden representation in the last time step $\mathbf{h}_T$ is further fed into the top classifier such as softmax classifier to perform classification task, which calculates the probability of a predicted label y_k among K classes as:

$$P(y_k|\mathbf{h}_T) = \frac{\exp\{\mathbf{W}_k^\top \mathbf{h}_T + b_k\}}{\sum_{i=1}^{K} \exp\{\mathbf{W}_i^\top \mathbf{h}_T + b_i\}} \tag{3}$$

3.2. Attention Module

We propose an attention-weighting mechanism to calculate the attention gate in Equation 1. To obtain a comprehensive summarization for each time step of input sequence and thereby achieve an accurate attention value for the degree of salience, we take advantage of bi-directional RNN. Specifically, the attention weight a_t at time step t is calculated by

$$a_t = \sigma(A(\overrightarrow{h}_t; \overleftarrow{h}_t) + b) \tag{4}$$

Herein, $\overrightarrow{h}_t$ and $\overleftarrow{h}_t$ are the hidden representations of bi-directional RNN model:

$$\overrightarrow{h}_t = g(\overrightarrow{\mathbf{W}}\mathbf{x}_t + \overrightarrow{\mathbf{U}}\overrightarrow{h}_{\mathbf{t-1}} + \overrightarrow{\mathbf{b}}) \tag{5}$$

$$\overleftarrow{h}_t = g(\overleftarrow{\mathbf{W}}\mathbf{x}_t + \overleftarrow{\mathbf{U}}\overleftarrow{h}_{\mathbf{t+1}} + \overleftarrow{\mathbf{b}}) \tag{6}$$

The softplus function $g(x) = \ln(1 + \exp(x))$ is used as the activation function g. The utilization of bi-directional RNN makes the model focus on current time steps and take into account the adjacent temporal information in both directions. Considering the simplicity of the model and the fact that we focus on modeling the distribution characteristics around the operated time step, plain-RNN is qualified for the job (preliminary experiments show that LSTM does not help here).

A sigmoid function is employed for σ in Equation 4 to bound the attention weight in $[0, 1]$. Apart from serving as the attention gate for the top hidden representation module to control the involved information flow, another important role the learned attention weights play is to provide a interpretability about the degree of salience of each time step, which can be potentially used for applications such as sequence salience detection or sequence noise filtering.

3.3. Parameter Learning

Given a training set $\mathcal{D} = \{(\mathbf{x}^{(n)}_{1,\ldots,T}, \mathbf{y}^{(n)})\}_{n=1,\ldots,N}$ containing N pairs of sequences of length T (here length T can differ from sample to sample) and their associated labels. We learn all the parameters Θ from three modules (i.e., attention module, hidden representation learning module and softmax classifier module) of the Recurrent Attention-gated model by minimizing the conditional negative log-likelihood of the training data with respect to the parameters:

$$\mathcal{L}(\Theta) = -\sum_{n=1}^{N} \log P\left(\mathbf{y}^{(n)} | \mathbf{x}^{(n)}_{1,\ldots,T}\right) \tag{7}$$

The model can be readily trained in an end-to-end manner. The loss function is back-propagated through top hidden representation module and attention model successively using back-propagation through time algorithm [38]. We employ RMSprop as the gradient descent optimization algorithm with gradient clipping between -5 and 5 [2].

3.4. Comparison with LSTM and GRU

Our Recurrent Attention-Gated Model is specially designed with salience detection in mind. It should be noted that our Model is similar to the design of the RNN variants like GRU and LSTM except for four key differences dedicated to the modeling of salience:

- We employ bi-directional RNN instead of single direction RNN to take into account both the preceding and the following information of the sequence in the attention-weighting mechanism. It helps to model the temporal smoothness of attention distribution. It should be noted that it is different from the design of the gates in the bi-directional LSTM model since the latter just concatenates the hidden representations of two directional LSTM, which does not remedy the downside that each gate of them is still calculated by considering only one-directional information.

- We only focus on one scalar attention score to measure the relevance of the current time step instead of generally modeling gate value for each hidden unit as done by GRU and LSTM. In this way, we can obtain an interpretable salience detection.

- We separate the attention modeling and recurrent hidden representation learning as two independent modules to decrease the degree of coupling. One of the advantage would be we can customize the specific recurrent structure for each modules with different complexity according to the requirements.

- Our model only contains one scalar gate, namely attention gate, rather than 2 vectorial gates in GRU and 3 gates in LSTM. Doing so enforces the attention gate to take full responsibility of modeling all the salience information and thereby maximize the discrimination. In addition, the model contains fewer parameters (compared to LSTM) and simpler gate structure with less redundancy (compared to GRU and LSTM) to train. It eases the training procedure and can somewhat alleviate the potential over-fitting and have better generalization given small amount of training data, which is demonstrated in section 4.1.3.

4. Experiments

We performed experiments with RAGN on three different tasks with three datasets of different modalities: (1) speech recognition with an audio dataset, (2) sentiment analysis with a text dataset, and (3) event recognition with a video dataset.

Hyper-parameter validation To make the effective region of the sigmoid function of RAGN model adaptive to the specific data, we validate the learning rate for parameters A and b in Equation 4. Larger learning rate leads to sharper distribution of attention weights, namely, augmenting on more important time steps and overlook smaller weights.

For all the recurrent networks mentioned in this work (RAGN, GRU, LSTM and plain-RNN), the number of hidden units is tuned by selecting the best configuration from the option set $\{64, 128, 256\}$ using validation set. The dropout value is validated from the option set $\{0.0, 0.25, 0.5\}$ to avoid the potential overfitting.

4.1. Preliminary Experiments with Synthetic Dataset

4.1.1 Dataset

We conduct preliminary experiments on a synthetic dataset constructed from the Arabic spoken digit dataset [9]. The Arabic spoken digit dataset contains 8800 utterances, which were collected by asking 88 Arabic native speakers to utter all 10 digits ten times. Each sequence consists of 26-dimensional MFCCs which were sampled at 11,025Hz, 16-bits using a Hamming window. We append the white noise to the beginning and the end of each sample to make them noisy. The length of the noise appended is randomized to ensure that the model does not learn to just focus on the middle of the sequence.

4.1.2 Experimental setup

We use the same data division asivision as Hammami and Bedda [9]: 6600 samples as training set and left 2200 samples as test set. We further set aside 1100 samples from

training set as the validation set. There is no subject overlap between the three sets.

We compare the performance of our Recurrent Attention-Gated Networks with three types of baseline models:

Attention module + NN. One straightforward way to perform the subsequent classification after attention calculation is to employ feed-forward neural network on the weighted sum of the different time steps:

$$\mathbf{v} = \sum_{t=1}^{T} a_t * \mathbf{x}_t$$
$$\mathbf{h} = g(\mathbf{W} \cdot \mathbf{v} + \mathbf{b}) \quad (8)$$

The obtained hidden representation $\mathbf{h}$ is further fed into the softmax classifier as TAGM.

State-of-the-art sequence classification models. HCRF and HULM are both graphical models extended from the conditional random fields (CRF [21]) by inserting hidden layers to model the non-linear latent structure in the data. The difference lies in the structure of hidden layers: HCRF use a chain of k-nomial latent variables while HULM utilizes k binary stochastic hidden units.

Recurrent neural networks. Since our model is a recurrent networks equipped with a gate mechanism, we compare it with other recurrent networks: plain-RNN, GRU, LSTM.

In the experiments of comparing the generalizability with the varying size of training data, we augment the size of training data from 1100 to 5500 increasingly to compare the performance of the three models. We select the best model configuration for hidden layers from the option set $\{64, 128, 256\}$ with the validation set.

4.1.3 Results

Comparison of generalizability with the varying size of training data. We first conduct experiments to investigate the generalizability of RAGN to GRU and LSTM by varying the size of training data on the noisy arabic dataset. Figure 3 presents the experimental results. It shows that RAGN exhibits better generalizability than GRU and LSTM when trained on a small amount of training data, which is consistent with the fact that it has fewer parameters to learn. GRU outperforms LSTM when training data size equals 2200 and 3300 which probably results from the less gates.

Sequence Salience Detection. In order to evaluate the performance of sequence salience detection by our Recurrent Attention model, we visualize the attention weights of our model trained on noisy arabic dataset, which is illustrated in Figure 4. It shows that the attention model can correctly detect the informative salience part from the raw dataset.

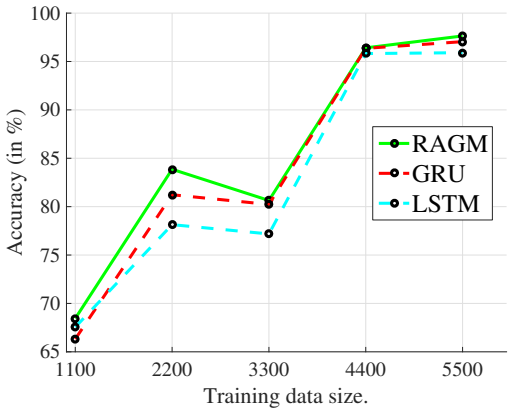

Figure 3. The classification accuracy on the Arabic speech dataset as a function of the size of training data.

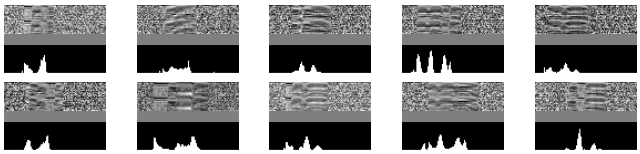

Figure 4. The visualization of attention weights of Recurrent Attention Model, with sigmoid weight $r = 1$. For each subfigure, top image is original feature space and bottom image is the distribution of attention weights. The x-axis is the time domain and the y-axis indicates the calculated attention weight.

To investigate the effect of the temporal information contained in the hidden representation, we also visualize the attention weight of the Attention module + Neural Network classifier, which is shown in Figure 5. It shows that Recurrent Attention model with attention gate would result in cleaner and more smooth weight distribution, which is mainly achieved by the temporal modeling in recurrent way.

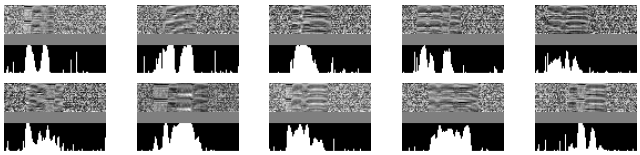

Figure 5. The weighted features are fed into Feed-forward Neural Networks. . For each subfigure, top image is original feature space and bottom image is the transformed feature space. The x-axis is the time domain and the y-axis indicates the calculated attention weight.

Evaluation of Classification Performance Table 1 presents the classification performance of several sequence classifiers on Arabic dataset. In order to the investigate the

effect of the manually added noise information, we perform experiments on both clean and noisy versions of data.

While Plain-RNN completely fails because of the negative interference by the noise, other three recurrent models with gate-setup do not suffer from the affect and obtain the comparable (even better by GRU and RAGN) performance with the state-of-the-art result achieved by HCRF on clean data. Our model achieves best result among all classifiers with single-directional recurrent configuration. GRU also obtains very good performance which is better than LSTM, this probably results from better generalization of our model and GRU compared to LSTM on the relatively small dataset due to the simpler gate setup. We also perform experiments with the bidirectional version of GRU, LSTM and TAGM, in which the performance of both Bi-GRU and Bi-LSTM are improved larger than our Bi-TAGM, which indicates that the bi-directional mechanism in the attention model can capture most bi-directional information. Our TAGM using 47 K parameters already achieves comparable result with the Bi-LSTM, Bi-GRU perform better than our Bi-TAGM with much larger amount of parameters (441 K).

The experiments on the synthetic dataset demonstrate that our model can actually capture the salient parts accurately. It exhibits better generalizability than LSTM when training on smaller amount of data. The bidirectional mechanism of TAGM enables it to achieve comparable performance with Bi-GRU and Bi-LSTM.

Table 1. Classification accuracy (%) on Arabic spoken dataset by different sequence classification models. Asterisked models ($*$) perform on the clean version of data. Herein, "AM-NN" denotes the Attention-weighting Module + Neural Network classifier while "TAGM"refers to our Temporal Attention-Gated Model. See text for details.

Model	H	|Parameters|	Accuracy
HULM* [25]	–	–	95.32
HCRF* [25]	–	–	96.32
HULM	–	–	
HCRF	–	–	
Plain-RNN*	256	75 K	94.95
Plain-RNN	256	75 K	10.95
GRU	128	61 K	97.05
LSTM	128	81 K	95.91
NN	64	2.4 K	65.50
AM-NN	128-64	43 K	85.59
TAGM	128-64	47 K	97.64
Bi-GRU	256	441 K	99.09
Bi-LSTM	128	162 K	97.86
Bi-TAGM	128-128	83 K	97.91

4.2. Sentiment Analysis

Sentiment analysis in language is a popular research topic in the field of natural language processing (NLP) which requires to consider not only the key words with strong sentiment but also the semantic compositionality between phrases. Hence our model is a good fit for this task, herein one sentence can be considered as a sequence of which each word corresponds to a time step.

4.2.1 Dataset

The Stanford Sentiment Treebank (SST) [34] is a data corpus of movie review excerpts. It consists of 11855 sentences each of which is assigned a score to indicate the sentimental attitude towards the movie reviews. 215,154 phrases are obtained from parsing all sentences by the Stanford Parser [19]. Both the sentence-level and phrase-level labels are provided with two resolutions: binary-classification task and fine-grained (5-class) task.

4.2.2 Experimental Setup

We utilize 300-d *Glove* word vectors pretrained over the Common Crawl [27] as the features for each word of the sentences. Our model is well suitable to perform sentiment analysis using sentence-level labels. Nevertheless, we also make a try with both the labels in two levels so as to have a fair and intuitive comparison with state-of-the-art baselines.

Follwing Socher et al. [34], we apply the fixed data division: 8544/1101/2210 samples are used for training, validation and test respectively while the corresponding splits are 6920/872/1821 in the binary classification task.

4.2.3 Results

Sequence Salience Detection In order to investigate the performance of salience detection by our model on SST data, we visualize the calculated attention weights for each word in the test sentences. Figure 6 presents a number of examples that are predicted correctly by our model in the binary-classification task. It shows that our model is able to successfully capture the key sentimental words and omit irrelevant words, even for the sentences with complicated syntax. We especially test the examples that are about the negated expression. As shown in the last two sentences in Figure 6, our model can deal with them very well. We also investigate the samples our model fails to predict. As shown in Figure 7, it seems that the strongly sentimental words would mislead our model in the sentences with very confusing context. In this case, it is very hard for the model to understand the intention hidden behind the words.

Figure 6. The visualization of attention weights of Recurrent Attention Model, all the samples listed are predicted correctly. 'Gray' colormap is employed in the visualization: the whiter the color is, the higher the weight value is. The scores displayed are the groundtruth label indicating the writer's overall intention for this review.

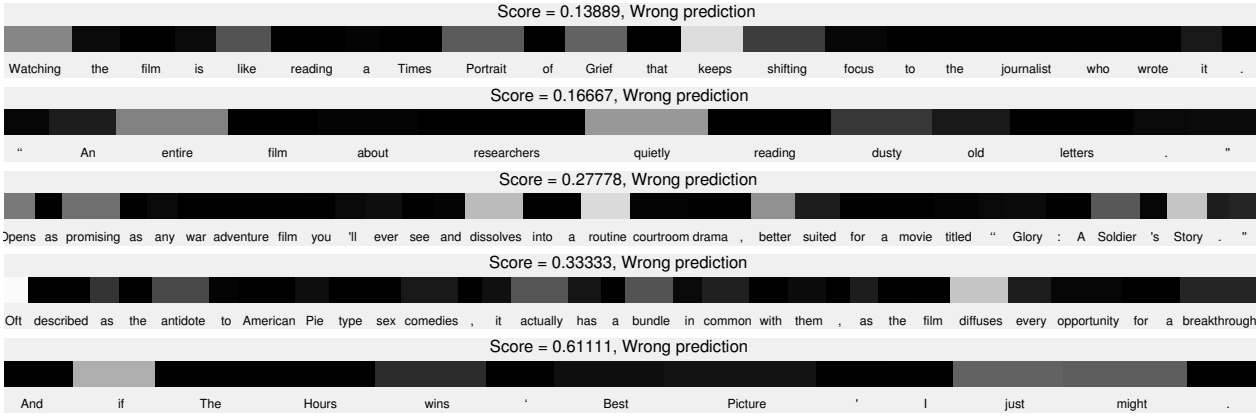

Figure 7. The visualization of attention weights of Recurrent Attention Model, all the samples listed are predicted Wrong.'Gray' colormap is employed in the visualization: the whiter the color is, the higher the weight value is. The scores displayed are the groundtruth label indicating the writer's overall intention for this review.

Evaluation of Classification Performance We conduct two sets of experiments to evaluate the performance of our model with the comparison to the baseline models. Since our model is designed for general noisy sequence modeling instead of syntax-oriented sentence modeling, it makes more sense to only use sentence-level labels, although phrase-level labels are also provided in SST dataset. Table 2 shows the experimental results of several sequential models with only sentence-level labels. Our model achieves the best result in both binary classification task and fine-grained (5-class) task. LSTM and GRU outperform plain-RNN model due to the information-filtering capability performed by additional gates. It is worth mentioning that our model achieves better performance than LSTM with only half amount of hidden parameters.

Table 2. Classification accuracy (%) on Stanford Sentiment TreeBank dataset by different models. We conduct experiments on both binary and fine-grained (5-class) classification tasks. Herein, all models are trained with **only sentence-level labels**. See text for details.

	Model	Binary	Fine-grained
Graphical models	HULM		
	HCRF		
Syntactic compositions	DAN-ROOT [13]	85.7	46.9
Recurrent models	Plain-RNN	83.9	42.3
	GRU	85.4	46.7
	LSTM	85.9	47.2
Our model	RAGN	86.2	48.0

Table 3. Classification accuracy (%) on Stanford Sentiment TreeBank dataset by different models. We conduct experiments on both binary and fine-grained (5-class) classification tasks. Herein, all models are trained with **both phrase-level and sentence-level labels**. See text for details.

	Model	Binary	Fine-grained
Unordered compositions	NBOW-RAND [13]	81.4	42.3
	NBOW [13]	83.6	43.6
	BiNB [13]	83.1	41.9
Syntactic compositions	RecNN [33]	82.4	43.2
	RecNTN [34]	85.4	45.7
	DRecNN [12]	86.6	49.8
	DAN [13]	86.3	47.7
	TreeLSTM [35]	86.9	**50.6**
	CNN-MC [18]	**88.1**	47.4
	PVEC [22]	87.8	48.7
Our model	RAGN	87.6	50.1

To have a fair comparison with the existing sentimental analysis models, we conduct the second set of experiments with both sentence-level and phrase-level labels. The results are presented in Table 3. It shows that our model outperforms most of the existing models and achieves comparable accuracy with the state-of-the-art result. It actually obtains an overall best results considering both binary and fine-grained cases. This is an encouraging result, in particular, since our model is not specifically designed towards sentiment analysis task.

4.3. event recognition

4.3.1 Dataset

Columbia Consumer Video (CCV) Database [17] is an unconstrained video database collected from YouTube without any post-editing. It consists of 9317 web videos with the average duration of 80 seconds (210 hours in total). Except some negative background videos, each video is manually annotated into one or more of 20 semantic categories such as 'basketball', 'ice skating', 'biking', 'birthday' and so on. It is a very challenging database due to much noise and irrelevant segments contained inside.

4.3.2 Experimental setup

Following Jiang et al [17], we use the fixed training/test division: 4659 videos as the training set and 4658 as the test set. We compare our model with the baseline method [16] on this dataset, which performs classification with SVM on the Bag-of-words representations of each of several popular features separately and then combines the results using late fusion. Its experimental results show that CNN features performs best among all features they tried, hence we choose to use CNN features with the same setup, i.e., the outputs (4,096 dimensions) of the seventh fully-connected layer of a pre-trained AlexNet model [20]. For the sake of computational efficiency, we extract CNN features with sampling rate 1/8 (one frame every eight).

Since there can be more than one event (correct label) happened in a sample and mean average precision (mAP) is typically used as the evaluation metric for CCV [17, 16], we perform binary classification for each category but train them jointly, hence the prediction score for each category is calculated by a sigmoid function instead of softmax equation 3:

$$P(y_k = 1|\mathbf{x}_{1,\dots,T}) = P(y_k = 1|\mathbf{h}_T) = \frac{1}{1+\exp\{-(\mathbf{W}_k^\top \mathbf{h}_T + b_k)\}} \quad (9)$$

and joint binary cross-entropy over K categories is minimized:

$$\mathcal{L}(\Theta) = -\sum_{n=1}^{N}\sum_{k=1}^{K}\Big[\log P(y_k = 1|\mathbf{x}_{1,\dots,T}) + \log(1 - P(y_k = 0|\mathbf{x}_{1,\dots,T}))\Big] \quad (10)$$

4.3.3 Results

Sequence Salience Detection Salience detection for CCV database is an extremely difficult but appealing task due to the quite intricate scene it may contains and relatively long duration of videos. We select some representative frames from a video and check the corresponding learned attention weights to gain the insight about the performance of salience detection of our model. Figure 8 shows some examples that our model correctly locates the salient subsequences by the attention weights. Our model is able to capture the relevant action, object or scene to the event, e.g., the action of riding bike for the event 'biking', cake for the event 'birthday' and baseball playground for the event 'baseball'. It is interesting to note that the frame with the score 0.42 in event 'baseball' achieves the high score probably because of the real-time screen in the top right corner.

Evaluation of Classification Performance We compare our model with the event recognition system [16]. Table 4 presents the performance of both models. Although it is not a fair comparison due to the fact that the baseline method employs the one-vs-all strategy to train a separate classifier for each event whereas our model train all events jointly in a single classifier. Our model still achieve an encouraging result.

Table 4. Mean Average Precision (mAP) of baseline and our model on CCV dataset. See text for details.

Model	Training strategy	Feature	mAP
BOW+SVM +late average fusion	Separately (one-vs-all)	SIFT	0.523
		STIP	0.449
		SIFT+STIP	0.551
		CNN	0.673
RAGN	Jointly	CNN	0.600

5. Conclusion

In this work, we presented the Recurrent Attention-Gated Networks (RAGN), a new model for the noisy sequence classification. The model combines the ideas from attention model and gated recurrent networks to detect the salient segments and filter out the noisy segments. The resulted hidden representation does not suffer from the effect from the noise and thereby improve the final classification performance. Furthermore, the learned attention scores provide a physically meaningful interpretation of relevance of each time step (subsequence) to the final decision, which has many potential applications such as sequence noise reduction and sequence key-frame detection. the results of experiments with RAGN on several tasks shows that our model performs well and is able to effectively locate the key frames (subsequences) of a sequence which is crucial to the final decision on various challenging applications.

This study is an initial investigation into modeling noisy sequence combining attention model and recurrent networks, and we foresee several extensions of this work in two aspects including w.r.t. the model and w.r.t. the application for future work. As to the extensions for model, we aim to incorporate the reinforcement learning in the learning of attention module. Specifically, the feedback of classification performance in current iteration of parameter update would affect the update of attention module in the next iteration in an explicit way, thus there could be a reward or feedback loop from the hidden representation back to the attention module. Concerning the extensions w.r.t. the application, our model can be potentially applied to abstract summarization in the NLP research, whose output is the sequential prediction instead of classification. In addition, document classification is also a suitable application for our model worth an attempt.

References

[1] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014. 2

[2] Y. Bengio, N. Boulanger-Lewandowski, and R. Pascanu. Advances in optimizing recurrent networks. In *ICASSP*, pages 8624–8628. IEEE, 2013. 4

[3] X. Chen and C. L. Zitnick. Mind's eye: A recurrent visual representation for image caption generation. In *CVPR*, pages 2422–2431. IEEE Computer Society, 2015. 2

[4] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *CoRR*, abs/1409.1259, 2014. 1, 2

[5] T.-M.-T. Do and T. Artieres. Neural conditional random fields. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9. JMLR: W&CP, 5 2010. 2

[6] H. Fang, S. Gupta, F. N. Iandola, R. K. Srivastava, L. Deng, P. Dollr, J. Gao, X. He, M. Mitchell, J. C. Platt, C. L. Zitnick, and G. Zweig. From captions to visual concepts and back. In *CVPR*, pages 1473–1482. IEEE Computer Society, 2015. 2

[7] A. Graves. Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850, 2013. 2

[8] A. Graves, S. Fernndez, and F. Gomez. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *In Proceedings of the International Conference on Machine Learning, ICML 2006*, pages 369–376, 2006. 2

[9] N. Hammami and M. Bedda. Improved tree model for Arabic speech recognition. In *Int. Conf. on Computer Science and Information Technology*, pages 521–526, 2010. 4, 5

[10] A. Haque, A. Alahi, and L. Fei-Fei. Recurrent attention models for depth-based person identification. In *The IEEE*

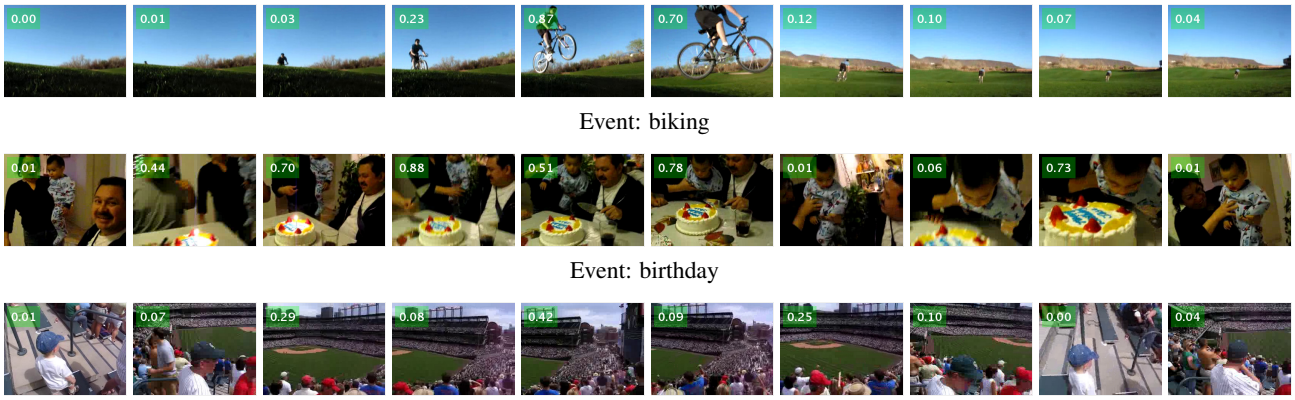

Figure 8. The calculated attention weights of Recurrent Attention-Gated Model for examples from test set of CCV database. The attention weight is indicated for selected representative frames.

Conference on Computer Vision and Pattern Recognition (CVPR), June 2016. 2

[11] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, Nov. 1997. 1, 2

[12] O. İrsoy and C. Cardie. Deep recursive neural networks for compositionality in language. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2096–2104. Curran Associates, Inc., 2014. 8

[13] M. Iyyer, V. Manjunatha, J. Boyd-Graber, and H. Daumé III. Deep unordered composition rivals syntactic methods for text classification. In *Association for Computational Linguistics*, 2015. 8

[14] T. Jaakkola, M. Diekhans, and D. Haussler. A discriminative framework for detecting remot protein homologies. *Journal of Computational Biology*, 7(1-2):95–114, 2000. 2

[15] L. A. Jeni, A. Lőrincz, Z. Szabó, J. F. Cohn, and T. Kanade. Spatio-temporal event classification using time-series kernel based structured sparsity. In *2014 European Conference on Computer Vision (ECCV)*, 2014. 2

[16] Y.-G. Jiang, Q. Dai, T. Mei, Y. Rui, and S.-F. Chang. Super fast event recognition in internet videos. *IEEE Transactions on Multimedia*, 17(8):1–13, 2015. 8, 9

[17] Y.-G. Jiang, G. Ye, S.-F. Chang, D. Ellis, and A. C. Loui. Consumer video understanding: A benchmark database and an evaluation of human and machine performance. In *Proceedings of ACM International Conference on Multimedia Retrieval (ICMR), oral session*, 2011. 8

[18] Y. Kim. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October 2014. Association for Computational Linguistics. 8

[19] D. Klein and C. D. Manning. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 423–430, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics. 6

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012. 8

[21] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. 2, 5

[22] Q. Le and T. Mikolov. Distributed representations of sentences and documents. In T. Jebara and E. P. Xing, editors, *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1188–1196. JMLR Workshop and Conference Proceedings, 2014. 8

[23] M.-T. Luong, H. Pham, and C. D. Manning. Effective approaches to attention-based neural machine translation. In *EMNLP*, 2015. 2

[24] T. Mikolov, S. Kombrink, L. Burget, J. Cernock, and S. Khudanpur. Extensions of recurrent neural network language model. In *ICASSP*, pages 5528–5531. IEEE, 2011. 2

[25] W. Pei, H. Dibeklioglu, D. M. J. Tax, and L. van der Maaten. Time series classification using the hidden-unit logistic model. *arXiv*, abs/1506.05085, 2015. 1, 2, 6

[26] J. Peng, L. Bo, and J. Xu. Conditional neural fields. In Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 1419–1427. Curran Associates, Inc., 2009. 2

[27] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014. 6

[28] A. Quattoni and T. Darrell. Latent-dynamic discriminative models for continuous gesture recognition. In *Proceedings of CVPR07*, pages 1–8, 2007. 2

[29] A. Quattoni, S. Wang, L.-P. Morency, M. Collins, and T. Darrell. Hidden conditional random fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(10):1848–1852, Oct. 2007. 1, 2

[30] L. R. Rabiner. Readings in speech recognition. chapter A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, pages 267–296. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1990. 2

[31] B. Sankaran, H. Mi, Y. Al-Onaizan, and A. Ittycheriah. Temporal attention model for neural machine translation. *CoRR*, abs/1608.02927, 2016. 2

[32] J. SCHMIDHUBER. A local learning algorithm for dynamic feedforward and recurrent networks. *Connection Science*, 1(4):403–412, 1989. 2

[33] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 151–161, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. 8

[34] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Stroudsburg, PA, October 2013. Association for Computational Linguistics. 6, 8

[35] K. S. Tai, R. Socher, and C. D. Manning. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1556–1566. The Association for Computer Linguistics, 2015. 8

[36] L. Theis and M. Bethge. Generative image modeling using spatial lstms. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, NIPS'15, pages 1927–1935, Cambridge, MA, USA, 2015. MIT Press. 2

[37] L. van der Maaten, M. Welling, and L. K. Saul. Hidden-unit conditional random fields. In G. J. Gordon, D. B. Dunson, and M. Dudk, editors, *AISTATS*, volume 15 of *JMLR Proceedings*, pages 479–488. JMLR.org, 2011. 2

[38] P. J. Werbos. Generalization of backpropagation with application to a recurrent gas market model. *Neural Networks*, 1, 1988. 4

[39] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In D. Blei and F. Bach, editors, *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 2048–2057. JMLR Workshop and Conference Proceedings, 2015. 2

[40] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville. Describing videos by exploiting temporal structure. In *Computer Vision (ICCV), 2015 IEEE International Conference on*. IEEE, 2015. 2