Temporal Attention-Gated Model for Robust Sequence Classification

Wenjie Pei[1], Tadas Baltrušaitis[2], David M.J. Tax[1] and Louis-Philippe Morency[2]
[1]Pattern Recognition Laboratory, Delft University of Technology
[2]Language Technologies Institute, Carnegie Mellon University
W.Pei-1@tudelft.nl, tbaltrus@cs.cmu.edu, D.M.J.Tax@tudelft.nl, morency@cs.cmu.edu

Abstract

Typical techniques for sequence classification are designed for well-segmented sequences which have been edited to remove noisy or irrelevant parts. Therefore, such methods cannot be easily applied on noisy sequences expected in real-world applications. In this paper, we present the Temporal Attention-Gated Model (TAGM) which integrates ideas from attention models and gated recurrent networks to better deal with noisy or unsegmented sequences. Specifically, we extend the concept of attention model to measure the relevance of each observation (time step) of a sequence. We then use a novel gated recurrent network to learn the hidden representation for the final prediction. An important advantage of our approach is interpretability since the temporal attention weights provide a meaningful value for the salience of each time step in the sequence. We demonstrate the merits of our TAGM approach, both for prediction accuracy and interpretability, on three different tasks: spoken digit recognition, text-based sentiment analysis and visual event recognition.

1. Introduction

Sequence classification is posed as a problem of assigning a label to a sequence of observations. Sequence classification models have extensive applications ranging from computer vision [17] to natural language processing [1]. Most existing sequence classification models are designed for well segmented sequences and do not explicitly model the fact that irrelevant (noisy) parts may be present in the sequence. To reduce the interference of these irrelevant parts, researchers will often manually pre-process the dataset to remove irrelevant subsequences. This manual pre-processing can be very time consuming and reduce applicability in real-world scenarios.

A popular approach for sequence classification is gated recurrent networks like Gated Recurrent Units (GRU) [4] and Long Short-Term Memory (LSTM) [11]. They employ gates (e.g., the input gate in the LSTM model) to balance between current and previous time steps when memorizing the temporal information flow. However, these vectorial gates are applied individually to each dimension of the information flow, thus it is hard to interpret the relative importance of the input time observations (i.e., time steps). What subset of sequential observations is the most salient for the classification task? Another way to balance the information flow, as we do in this work, is the adoption of attention-based mechanism, which applies individual attention scores to each observation (time step), allowing for better interpretability.

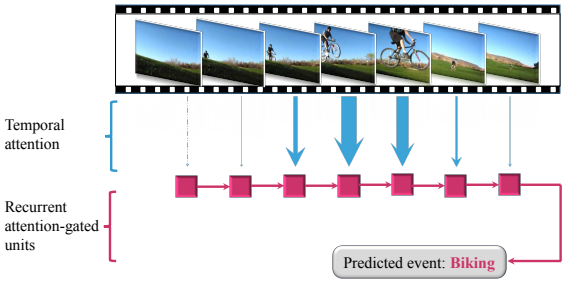


Figure 1. Our proposed model first employs an attention module to extract the salient frames from the noisy raw input sequences, and then learns an effective hidden representation for the top classifier. The wider the arrow is, the more the information is incorporated into the hidden representation. The dashed line represents no transfer of information.

In this paper, we introduce the Temporal Attention-Gated Model (TAGM) which extends the idea of attention-based mechanism to sequence classification tasks (see overview in Figure 1). TAGM's attention module automatically localizes the salient observations which are relevant to the final decision and ignore the irrelavant (noisy) parts of the input sequence. We created a new recurrent neural unit that can learn a better sequence hidden representation based on the attention scores. Consequently, TAGM's classification decision is made based on the selected relevant segments, improving accuracy over the conventional models that take into account the whole input sequence.

Notably, compared to conventional sequence classification models, TAGM benefits from the following advantages:

- It is able to automatically capture salient parts of the input sequences thereby leading to better performance.
- The inferred attention (scalar) scores provide a meaningful interpretation for the informativeness of each observation in the sequence.
- 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.
- The proposed model is able to generalize to tasks in computer vision, speech recognition, and natural language processing.

2. Related Work

While a full review of previous sequence classification models is beyond the scope of this paper, in this section we summarize approaches most relevant to our proposed approach, grouping them in three areas: sequence classification, attention models and recurrent networks.

Sequence Classification. The conventional sequence classification models can be divided roughly into two categories: generative and discriminative models.

The first category focuses on learning an effective intermediate representation based on generative models. These methods are typically based on the Hidden Markov Models (HMMs) [31]. The HMM is a generative model which can be extended to class-conditional HMMs for sequence classification by combining class priors via Bayes' rule. HMM can also be used as the base model for Fisher Kernel [14] to learn a sequence representation.

The second category is the discriminative graphical models which model the distribution over all class labels conditioned on the input data. Conditional random fields (CRF) [20] are discriminative models for sequence labeling which aims to assign one label for each sequence observation. A potential drawback of common CRFs is that the linear mapping between observations and labels cannot model complex decision boundaries, which gives rise to many non-linear CRF-variants (e.g., latent-dynamic CRFs [29], conditional neural fields [27], neural conditional random fields [5] and hidden-unit CRF model [39]). Hidden-state CRF (HCRF) [30] employs a chain of k-nomial latent variables to model the latent structure and has been successfully used in the sequence labeling. Similarly, hidden unit logistic model (HULM) [26] utilizes binary stochastic hidden units to represent the exponential hidden states so as to model more complex latent decision boundaries.

Aforementioned works are specifically designed for well segmented sequences and hence cannot cope well with noisy or unsegmented sequences.

Attention Models. Inspired by the attention scheme of human foveal vision, attention model was 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 can also result in better interpretability [41]. Attention models have been applied to image and video captioning [41, 3, 6, 42], machine translation [1, 22, 32], depth-based person identification [10] and speech recognition [8]. To the best of our knowledge, our TAGM is the first end-to-end recurrent neural network to employ the attention mechanism in the temporal domain of sequences, with the added advantage of interpretability of its temporal salience indicators (i.e., temporal attention) at each time step (sequence observation). Our work is different from prior work focused on spatial domain (e.g., images) such as the model proposed by Sharma et al. [34].

Recurrent Networks. Recurrent Neural Networks (RNN) learn a representation for each time step by taking into account both the observation at current time step and the representation in the previous one [33]. The biggest advantage of recurrent neural networks lies in their capability of preserving information over time by the recurrent mechanism. Recurrent networks have been successfully applied to various tasks including language modeling [23], image generation [38] and online handwriting generation [7]. To address the gradient vanishing problem of plain-RNN when dealing with long sequences, LSTM [11] and GRU [4] were proposed. They are equipped with the gates to balance the information flow from the previous time step and current time step dynamically. Inspired by this setup, our TAGM model also employs a gate to filter out the noisy time steps and preserve the salient ones. The difference from the LSTM and GRU is that the gate value in our model is fed from the attention module which focuses on learning the salience at each time step.

3. Temporal Attention-Gated Model

Given as input an unsegmented sequence of possibly noisy observations, our goal is to: (1) calculate a salience score for each time step observation in our input sequence, and (2) construct a hidden representation based on the salience scores, best suited for the sequence classification task. To achieve these goals, we propose the Temporal Attention-Gated Model (TAGM) which consists of two modules: temporal attention module, and recurrent attention-gated units. Our TAGM model can be trained in an end-to-end manner efficiently. The graphical structure of the model is illustrated in Figure 2.

3.1. Recurrent Attention-Gated Units

The goal of the recurrent attention-gated units is to learn a hidden sequence representation which integrates the attention scores (inferred from the temporal attention module that will be discussed in the next section). In order to inte-

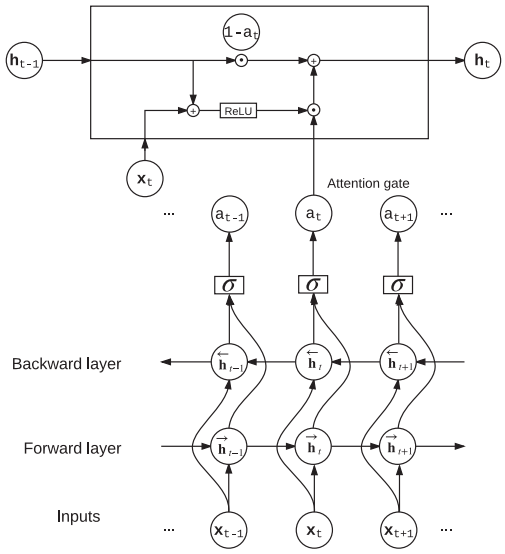


Figure 2. The graphical representation of our Temporal Attention-Gated Model (TAGM). The top part of the figure is the Recurrent Attention-Gated Units and the bottom is the Temporal Attention Module. Note that a_t is the saliency score represented as a scalar value instead of a vector, hence $\odot$ in the figure means multiplication between a scalar and a vector.

grate the attention scores in the recurrent network units, we define an attention gate to control how much information is incorporated from the input of the current time step based on the salience and relevance to the final task.

Formally, given an input 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 score at time step t is denoted as a_t, which is a scalar value that indicates the salience of current time step to the final decision. For this purpose, we define our core recurring process where the hidden state $\mathbf{h}_t$ at time step t is modeled as a convex summation:

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

Wherein, $\mathbf{h}_{t-1}$ is the previous hidden state and $\mathbf{h'}_t$ is the candidate hidden 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, $\mathbf{W}$ and $\mathbf{U}$ are respectively the linear transformation parameters for previous and current time steps while $\mathbf{b}$ is the bias term. We use the rectified linear unit (ReLU)[24] as the activation function g. Equation 1 uses attention score a_t to balance the information flow between current candidate hidden state $\mathbf{h'}_t$ and previous hidden state $\mathbf{h}_{t-1}$. High attention value will push the model to focus more on the current hidden state $\mathbf{h'}_t$ and input feature $\mathbf{x}_t$, while low attention value would make the model ignore the current input feature and inherit more information from previous time steps.

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

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

where $\mathbf{W}_i^\top$ and b_i refer to the parameters calculating the linear mapping score for the i-th class.

3.2. Temporal Attention Module

The goal of this module is to estimate the saliency and relevance of each sequence observation. This saliency score should not only be based on the input observation at the current time step, but also take into consideration information from neighboring observations in both directions. To model this neighborhood influence, we infer the attention score a_t in Equation 1 using a bi-directional RNN:

$$a_t = \sigma(\mathbf{m}^\top(\overrightarrow{h}_t; \overleftarrow{h}_t) + b) \tag{4}$$

Herein, $\mathbf{m}$ is the weight vector of our fusion layer which integrates both directional layers of our bi-directional RNN and b is the bias term. A sigmoid function is employed as the activation function σ at the top layer of the attention module in Equation 4 to constraint the attention weight to lie between $[0, 1]$. $\overrightarrow{h}_t$ and $\overleftarrow{h}_t$ are the hidden representations of a 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 ReLU functions are used as the activation functions g. Our choice of using plain bi-directional RNN model is motivated by the design goal of reducing the number of parameters in our model.

The learned attention weights a_t serve as the attention gate for Recurrent Attention-Gated Units to control the involved information flow. Furthermore, another important role the learned attention weights play is to provide an interpretability about the degree of salience of each time step.

3.3. End-to-End Parameter Learning

Suppose we are given a training set $\mathcal{D} = \{(\mathbf{x}^{(n)}_{1,\ldots,T}, y^{(n)})\}_{n=1,\ldots,N}$ containing N sequences of length T and their associated labels $y^{(n)}$. $\mathbf{x}^{(n)}_t \in \mathbb{R}^D$ denotes the observation at the t-th time step of the n-th sample and T can differ from sequence to sequence. We learn jointly the two TAGM modules (temporal attention module and recurrent attention-gated units) and the final sequence classifier by minimizing the conditional negative

log-likelihood of the training data with respect to the parameters:

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

Since all three modules (including the final sequence classifier) are analytically differentiable, our TAGM model can be readily trained in an end-to-end manner. The loss is back-propagated through top recurrent attention-gated units and temporal attention module successively using back-propagation through time algorithm [40].

3.4. Comparison with LSTM and GRU

While our model is similar to RNN variants like GRU and LSTM, it is specifically designed with salience detection in mind and has four key differences when compared to them:

- We only focus on one scalar attention score to measure the relevance of the current time step instead of generally modeling gate's multi-dimensional values for each hidden unit as done by GRU and LSTM. In this way, we can obtain an interpretable salience detection (demonstrated on three tasks in Section 4).
- We separate the attention modeling and recurrent hidden representation learning as two independent modules to decrease the degree of coupling. One of the advantages of this is our ability to customize the specific recurrent structure for each module with different complexity according to the requirements (eg., different size of hidden units in two modules of TAGM in Table 1).
- We employ a bi-directional RNN to take into account both the preceding and the following information of the sequence in the temporal attention module. It helps to model the temporal smoothness of the sequence of salience scores (demonstrated in Figure 4). 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 unidirectional LSTMs, which does not remedy the downside that all vectorial gates are still calculated by considering only one-directional information.
- Our model only contains one scalar gate, namely the 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. In addition, the model contains fewer parameters (compared to LSTM) and simpler gate structure with less redundancy (compared to GRU and LSTM). It eases the training procedure and can alleviate the potential over-fitting and has better generalization given small amount of training data, which is demonstrated in Section 4.1.3.

4. Experiments

We performed experiments with TAGM on three publicly available datasets , selected to show generalization across different tasks and modalities: (1) speech recognition on an audio dataset, (2) sentiment analysis on a text dataset, and (3) event recognition on a video dataset.

Experimental setup shared across experiments. For all the recurrent networks mentioned in this work (TAGM, 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 a validation set. The dropout value is validated from the option set $\{0.0, 0.25, 0.5\}$ to avoid potential overfitting. We employ RMSprop as the gradient descent optimization algorithm with gradient clipping between -5 and 5 [2].

We validate the learning rate for parameters $\mathbf{m}$ and b in Equation 4 to make the effective region of the sigmoid function of TAGM model adaptive to the specific data. Larger learning rate leads to sharper distribution of attention weights. Code reproducing the results of our experiments is available [1].

4.1. Speech Recognition Experiments

We first conduct preliminary experiments on a modified dataset constructed from the Arabic spoken digit dataset [9] to (1) evaluate the effectiveness of the two main modules of TAGM; (2) compare the generalizability of three different gate-setup recurrent models (TAGM, GRU and LSTM) with the varying size of the training data.

4.1.1 Dataset

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 13-dimensional Mel Frequency Cepstral Coefficents (MFCCs) which were sampled at 11,025Hz, 16-bits using a Hamming window. We append white noise to the beginning and the end of each sample to simulate the problem with unsegmented sequences. The length of the unrelated sub-sequences before and after the original audio clips 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 as Hammami and Bedda [9]: 6600 samples as training set and 2200 samples as test set. We further set aside 1100 samples from training set as the validation set. There is no subject overlap in the three sets.

We compare the performance of our TAGM with three types of baseline models:

Attention Module + Neural Network (AM-NN). To study the impact of our recurrent attention-gated unit, we include a baseline model which employs a feed-forward network

[1] https://github.com/wenjiepei/TAGM

directly on top of the temporal attention module. In this AM-NN model, $\mathbf{v}$ is defined as the weighted sum of input features:

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

Sequence classification is performed by passing $\mathbf{h}$ into a softmax layer, as done for our TAGM (see Equation 3).

Discriminative Graphical Models. HCRF and HULM are both extensions of CRF [20] by inserting hidden layers to model the non-linear latent structure in the data. The difference lies in the structure of hidden layers: HCRF uses a chain of k-nomial latent variables while HULM utilizes k binary stochastic hidden units.

Recurrent neural networks. Since our model is a recurrent network equipped with a gate mechanism, we compare it with other recurrent networks: plain-RNN, GRU, LSTM. We also investigate the bi-directional variant of our TAGM model (referred as Bi-TAGM), which employs the bi-directional recurrent configuration in the recurrent attention-gated units.

In our experiments, we also evaluate the generalizability when varying size of training data: from 1,100 to 5,500 training samples. During these experiments, the optimal configuration is selected automatically during validation from the option set {64,128,256}.

4.1.3 Results and Discussion

Evaluation of Classification Performance Table 1 presents the classification performance of several sequence classifiers on Arabic dataset. In order to investigate the effect of the manually added noise information, we perform experiments on both clean and noisy versions of data.

While the Plain-RNN is unable to recognize spoken digits in a noisy setting, other three recurrent models with gate-setup do not suffer from the noise and obtain comparable performance with the result achieved by HCRF on clean data. Our model achieves the best results among all classifiers with single-directional recurrent configuration. This probably results from better generalization of our model on the relatively small dataset due to the simpler gate setup and also the attention mechanism. We also perform experiments with the bi-directional version of GRU, LSTM and TAGM, in which our Bi-TAGM performs best. Bi-GRU achieves its best performance with 64 hidden units. It is worth mentioning that our (single-directional) TAGM using 47 K parameters already achieves comparable result with the Bi-LSTM and Bi-GRU, which indicates that the bi-directional mechanism in the attention module of TAGM enables it to capture most bi-directional information in the attention layer alone.

Comparison of generalizability with the varying size of training data. We first conduct experiments to compare the generalizability of TAGM to GRU and LSTM by varying the size of training data on the noisy Arabic dataset. Figure 3 presents the experimental results. It can be seen that TAGM exhibits better generalizability than GRU and LSTM on smaller training data sizes, which we believe is caused by the need to learn fewer model parameters, avoiding overfitting.

Table 1. Classification accuracy (%) on Arabic spoken digit dataset by different sequence classification models. Asterisked models ($*$) are trained and evaluated on the clean version of data. Note that we can customize separately the complexity of TAGM's two modules. This design advantage is shown when looking at the optimal TAGM model (after validation) which has 128 dimensions for the Temporal Attention Module, and 64 dimensions for the Recurrent Attention-Gated Units.

Model	#Hidden units	#Parameters	Accuracy
HULM* [26]	–	–	95.32
HCRF* [26]	–	–	96.32
HULM	–	–	88.27
HCRF	–	–	90.41
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	64	37 K	97.68
Bi-LSTM	256	587 K	97.45
Bi-TAGM	128-128	83 K	**97.91**

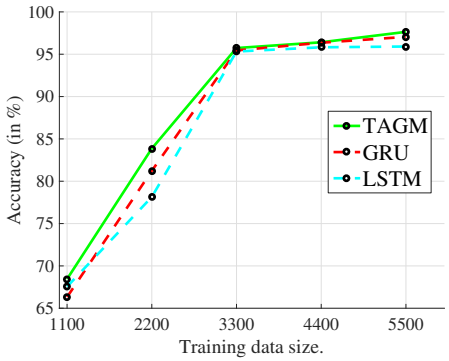
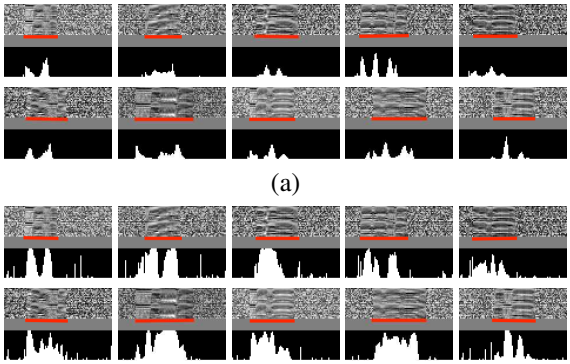


Figure 3. The classification accuracy on the noisy Arabic speech dataset as a function of the size of training data. Note that our TAGM model outperforms GRU and LSTM when less training data is available.

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

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 4.b. It shows that the



(a) (b)

Figure 4. The visualization of attention weights of TAGM in Figure a and Attention module+NN in Figure b (the weighted features are fed into Feed-forward Neural Networks) on 10 samples (one sample for each digit). For each subfigure, the top subplot shows the spectrogram of the original sequence data, the bottom subplot shows the attention values a_t over time. The red lines indicate the ground-truth of salient segments. Note that TAGM attention weights result in a cleaner attention representation.

TAGM results in a cleaner and smoother attention weight profile, also notice the spiky behavior, which is mainly achieved by the bi-directional RNN in our temporal attention module.

4.2. Sentiment Analysis Experiments

Sentiment analysis is a popular research topic in the field of natural language processing (NLP) which aims to identify the viewpoint(s) underlying a text span [25]. We conduct experiments for sentiment analysis to evaluate the performance of our TAGM model on the text modality.

4.2.1 Dataset

The Stanford Sentiment Treebank (SST) [36] is a data corpus of movie review excerpts. It consists of 11,855 sentences each of which is assigned a score to indicate the sentimental attitude towards the movie reviews. The dataset offers two types of annotations, sentiment annotations at the sentence level (with a total of 11,855 sentences) and at the phrase level (with a total of 215,154 phrases). The sentence-level and phrase-level labels are provided with two resolutions: binary-classification task (positive or negative) and fine-grained task (5-level classes).

4.2.2 Experimental Setup

Following previous work [36], we utilize 300-d *Glove* word vectors (300 dimensions) pretrained over the Common Crawl [28] as the features for each word of the sentences. Our model is well suited to perform sentiment analysis using sentence-level labels. Nevertheless, we also perform experiments with phrase-level labels so as to have a fair and intuitive comparison with state-of-the-art baselines.

Table 2. Classification accuracy (%) on Stanford Sentiment TreeBank dataset when training with only the sentence-level labels. We conduct experiments on both binary and fine-grained (5-class) classification tasks. Note that our model outperforms all others in the task.

	Model	Binary	Fine-grained
Graphical models	HULM	81.3	44.1
	HCRF	84.8	45.3
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	TAGM	**86.2**	**48.0**

Table 3. Classification accuracy (%) on Stanford Sentiment TreeBank dataset when training with both phrase-level and sentence-level labels. Our TAGM achieves the best overall result.

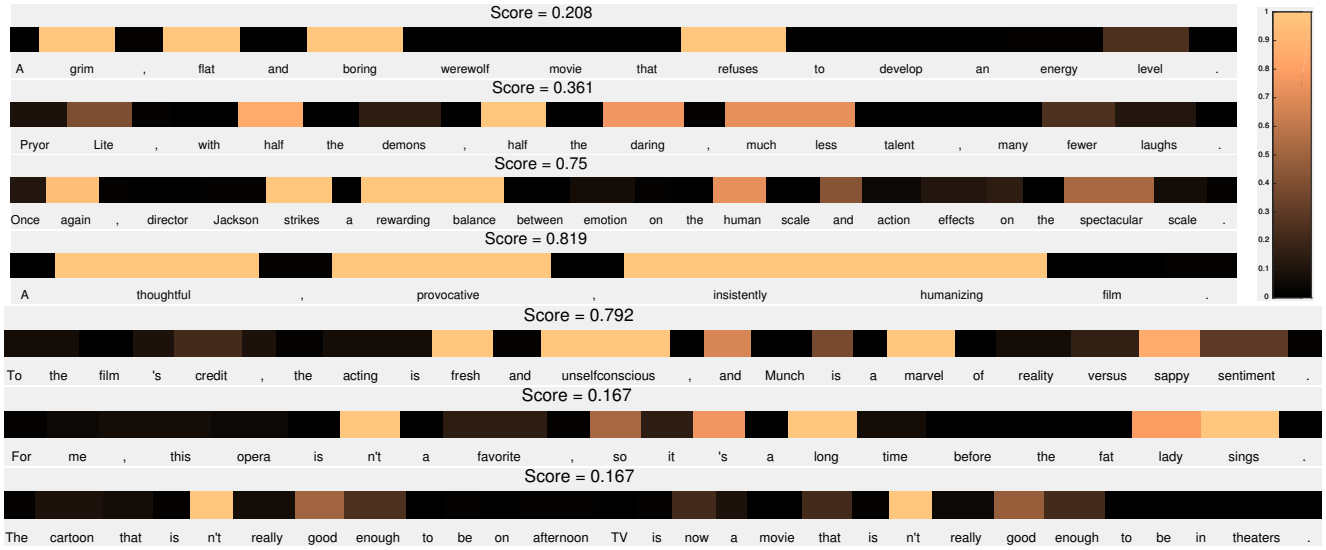
	Model	Binary	Fine-grained	Overall Performance
Unordered compositions	NBOW-RAND [13]	81.4	42.3	123.7
	NBOW [13]	83.6	43.6	127.2
	BiNB [13]	83.1	41.9	125.0
Syntactic compositions	RecNN [35]	82.4	43.2	125.6
	RecNTN [36]	85.4	45.7	131.1
	DRecNN [12]	86.6	49.8	136.4
	DAN [13]	86.3	47.7	134.0
	TreeLSTM [37]	86.9	**50.6**	137.5
	CNN-MC [18]	**88.1**	47.4	135.5
	PVEC [21]	87.8	48.7	136.5
Our model	TAGM	87.6	50.1	**137.7**

We follow the same data split as described by Socher et al. [36]: 8544/1101/2210 samples are used for training, validation and testing respectively in the 5-class task. The corresponding splits in the binary classification task are 6920/872/1821.

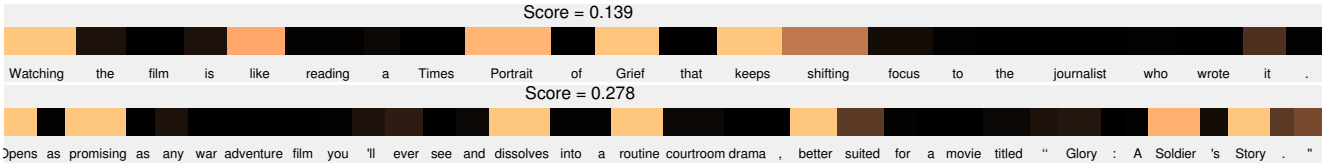
4.2.3 Results and Discussion

Evaluation of Classification Performance We conduct two sets of experiments to evaluate the performance of our model in comparison with the baseline models. Since our model is designed for unsegmented and possibly noisy sequences modeling, it is more suitable 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 trained 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 the hidden parameters.

To have a fair comparison with the existing sentiment 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 outper-



(a) Correct predictions.



(b) Wrong predictions.

Figure 5. The visualization of attention weights of Recurrent Attention Model: (a) correct predictions and (b) wrong predictions. The scores displayed are the groundtruth label indicating the writer's overall sentiment for this review. Darker color indicates smaller scores.

forms most of the existing models and achieves comparable accuracy with the state-of-the-art results. Our TAGM model actually obtains 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 NLP tasks.

Sequence Salience Detection In order to investigate the performance of salience detection by our TAGM model on Sentiment dataset (SST), we visualize the calculated attention weights for each word in the test sentences. Group (a) in Figure 5 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 also test the examples that include negated expressions. As shown in the last two sentences of group (a), our model can deal with them very well. We also investigate the samples our model fails to predict the correct sentiment label (see Figure 5b).

4.3. Event recognition Experiments

We subsequently conduct experiments for video event recognition to evaluate our model on the visual modality.

4.3.1 Dataset

Columbia Consumer Video (CCV) Database [16] is an unconstrained video database collected from YouTube videos without any post-editing. It consists of 9317 web videos with average duration of 80 seconds (210 hours in total). Except for 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 the many noisy and irrelevant segments contained inside these videos.

4.3.2 Experimental Setup

Following Jiang et al [16], we use the same split for training and test sets: 4659 videos as the training set and 4658 as the test set. We compare our model with the baseline method [15] on this dataset, which performs classification separately with Support Vector Machine (SVM) models trained on the bag-of-words representations for several popular features separately and then combines the results using late fusion. Its experimental results show that Convolutional Neural networks (CNNs) features perform 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 [19]. For the sake of computational efficiency, we extract CNN features with a sampling rate 1/8

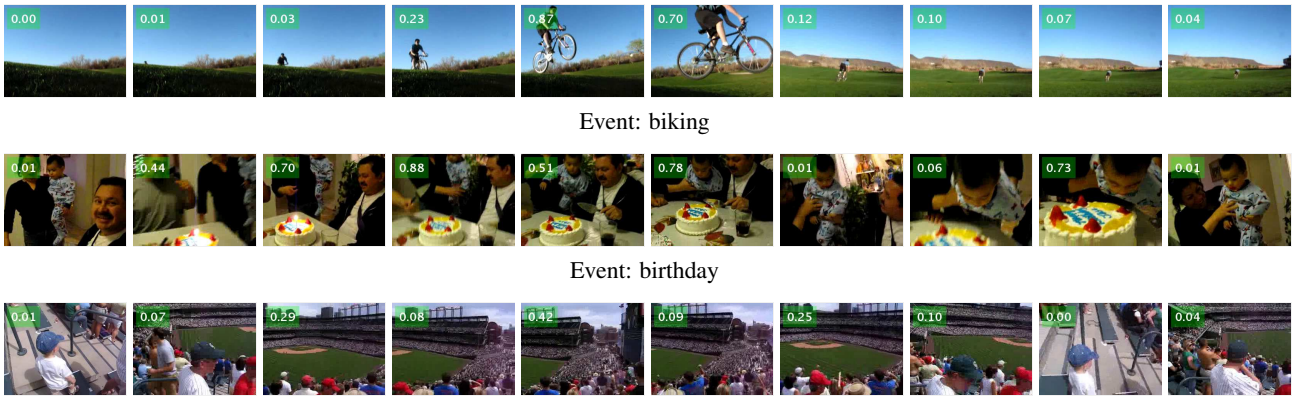


Figure 6. The calculated attention weights of our TAGM model for examples from test set of CCV database. The attention weight is indicated for representative frames. Our TAGM is able to capture the action of 'riding bike' for the event 'biking', 'cake' for the event 'birthday' and 'infield zone' for 'baseball'. A video containing these three complete sample sequences is presented in the supplementary material.

(one out of every eight frame).

We adopt mean Average Precision (mAP) as the evaluation metric, which is typically used for CCV dataset [16, 15]. Since more than one event (correct label) can happen in a sample, 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:

$$P(y_k = 1|\mathbf{h}_T) = \frac{1}{1 + \exp\{-(\mathbf{W}_k^\top \mathbf{h}_T + b_k)\}} \tag{9}$$

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

$$\mathcal{L} = -\sum_{n=1}^{N}\sum_{k=1}^{K}\Big[\log P(y_k = 1|\mathbf{h}_T) + \log(1 - P(y_k = 0|\mathbf{h}_T))\Big]$$

4.3.3 Results and Discussion

Evaluation of Classification Performance. We compare our model with the event recognition system proposed by dataset authors [15]. Table 4 presents the performance of several models for event recognition, in which our TAGM outperforms the other recurrent models by a large margin. The baseline BOW+SVM employs the one-vs-all strategy to train a separate classifier for each event while our model trains all events jointly in a single classifier. Our model still shows encouraging results since it is quite a challenging task for TAGM to capture salient sections for 20 events with complex scenes simultaneously. Moreover, our TAGM can provide a meaningful interpretation which the baseline models cannot do.

Sequence Salience Detection. Salience detection for CCV database is a difficult but appealing task due to complex and long scenes in videos. Figure 6 shows some examples where TAGM correctly locates the salient subsequences by the attention weights. Our model is able to capture the relevant action, object and 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.

Table 4. Mean Average Precision (mAP) of our TAGM model and baseline models on CCV dataset.

Model	Training strategy	Feature	mAP
BOW+SVM +late average fusion	Separately (one-vs-all)	SIFT	0.52
		STIP	0.45
		SIFT+STIP	0.55
		CNN	0.67
Plain-RNN	Jointly	CNN	0.45
GRU	Jointly	CNN	0.56
LSTM	Jointly	CNN	0.55
TAGM	Jointly	CNN	0.63

5. Conclusion

In this work, we presented the Temporal Attention-Gated Model (TAGM), a new model for classifying noisy and unsegmented sequences. The model is inspired by attention models and gated recurrent networks and is able to detect salient parts of the sequence while ignoring irrelevant and noisy ones. The resulting hidden representation suffers less from the effect of noise and and thus leads to better performance. Furthermore, the learned attention scores provide a physically meaningful interpretation of relevance of each time step observation for the final decision. We showed the generalization of our approach on three very different datasets and sequence classification tasks. As future work, our model could be extended to help with document or video summarization.

References

[1] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015.

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

[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.

[4] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio. On the properties of neural machine translation: Encoder-decoder approaches. In *Proceedings of SSST@EMNLP 2014, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, 2014.

[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.

[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.

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

[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.

[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.

[10] A. Haque, A. Alahi, and L. Fei-Fei. Recurrent attention models for depth-based person identification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

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

[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.

[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.

[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.

[15] 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.

[16] 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)*, 2011.

[17] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.

[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. Association for Computational Linguistics, 2014.

[19] 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. 2012.

[20] 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*, pages 282–289, 2001.

[21] 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.

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

[23] 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.

[24] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In J. Frnkranz and T. Joachims, editors, *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814. Omnipress, 2010.

[25] B. Pang and L. Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.

[26] W. Pei, H. Dibeklioğlu, D. M. J. Tax, and L. van der Maaten. Multivariate time-series classification using the hidden-unit logistic model. *IEEE Transactions on Neural Networks and Learning Systems*, 2017.

[27] 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. 2009.

[28] 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.

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

[30] 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.

[31] 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.

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

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

[34] S. Sharma, R. Kiros, and R. Salakhutdinov. Action recognition using visual attention. *CoRR*, abs/1511.04119, 2015.

[35] 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.

[36] 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.

[37] 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.

[38] 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.

[39] 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.

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

[41] 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.

[42] 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.