# Pooling Acoustic and Lexical Features for the Prediction of Valence

Zakaria Aldeneh
University of Michigan, Ann Arbor, USA
aldeneh@umich.edu

Soheil Khorram
University of Michigan, Ann Arbor, USA
khorrams@umich.edu

Dimitrios Dimitriadis
IBM T. J. Watson Research Center, USA
dbdimitr@us.ibm.com

Emily Mower Provost
University of Michigan, Ann Arbor, USA
emilykmp@umich.edu

## ABSTRACT

In this paper, we present an analysis of different multimodal fusion approaches in the context of deep learning, focusing on pooling intermediate representations learned for the acoustic and lexical modalities. Traditional approaches to multimodal feature pooling include: concatenation, element-wise addition, and element-wise multiplication. We compare these traditional methods to outer-product and compact bilinear pooling approaches, which consider more comprehensive interactions between features from the two modalities. We also study the influence of each modality on the overall performance of a multimodal system. Our experiments on the IEMOCAP dataset suggest that: (1) multimodal methods that combine acoustic and lexical features outperform their unimodal counterparts; (2) the lexical modality is better for predicting valence than the acoustic modality; (3) outer-product-based pooling strategies outperform other pooling strategies.

## CCS CONCEPTS

• **Computing methodologies** → **Neural networks**;

## KEYWORDS

Affective computing, multimodal emotion recognition, speech processing, deep learning

**ACM Reference Format:**
Zakaria Aldeneh, Soheil Khorram, Dimitrios Dimitriadis, and Emily Mower Provost. 2017. Pooling Acoustic and Lexical Features for the Prediction of Valence. In *Proceedings of 19th ACM International Conference on Multimodal Interaction (ICMI'17).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3136755.3136760

## 1 INTRODUCTION

Human communication consists of linguistic and paralinguistic cues [22]. The linguistic elements of speech encode the message (e.g., words) that a speaker is communicating, while the paralinguistic elements encode how the message is expressed. Paralinguistic elements can alter the meaning of a message (e.g., statement vs. question) or convey emotions (e.g., excitement, anger, etc.). Consequently, emotion recognition systems should consider both of these elements. In this paper, we explore deep learning architectures for multimodal speech emotion recognition that use both linguistic and paralinguistic features. Conventionally, multimodal fusion in deep learning uses pooling techniques to combine representations from different modalities to form a joint multimodal representation [9]. However, it is unclear which pooling technique is most effective for combining acoustic and lexical feature for the task of valence prediction. In this work, we investigate different pooling strategies that can be used to combine information from these two modalities.

Previous work showed that systems that incorporate both acoustic and lexical features are more accurate than those that only incorporate features from one modality [2, 12]. Traditionally, these multimodal approaches rely upon either early-fusion or late-fusion [19]. In late-fusion, a model is independently built for each modality, and decisions are generated from these independent models. These decisions are then combined to make a final decision. In early-fusion, multimodal feature vectors are created by combining the feature vectors from each modality. These augmented feature vectors are then used to learn a model. Early-fusion allows a model to consider low-level interactions between features from multiple modalities when making a prediction. However, these approaches assume a level of temporal synchrony between the individual modalities, which may not be valid. This is in contrast to late-fusion, where individual models consider features from only one modality, obscuring time-varying properties but alleviating the assumption of time-synchrony.

In this work, we investigate approaches for pooling representations from the acoustic and lexical modalities in neural networks for the end goal of making valence predictions. The pooling strategies that we investigate include element-wise summation, element-wise multiplication, concatenation, and outer-product. In addition, we also experiment with the multimodal compact bilinear pooling (CBP) approach [9], which provides a method for reducing the number of parameters obtained from a regular outer-product. Outer-product-based methods for pooling features allow the model to consider more expressive interactions between the features from the two modalities [9]. This is due to the fact that taking the outer-product allows all pairs of features from the two vectors to interact. Such methods showed success in computer vision applications [9, 15], but their use has not been investigated in linguistic/paralinguistic tasks.


*ICMI'17, November 13–17, 2017, Glasgow, UK*

ACM ISBN 978-1-4503-5543-8/17/11…$15.00
https://doi.org/10.1145/3136755.3136760

## 2 RELATED WORK

Li et al. [14] and Poria et al. [19] used models that were trained independently on different modalities as feature extractors. Li et al. applied a maximum entropy classifier to predict the speakers' stance in idealogical debates given lexical and acoustic features extracted from separately trained models. Poria et al. used lexical features extracted from a convolutional neural network along with manually extracted acoustic and visual features to perform multimodal sentiment predictions using a multiple kernel learning (MKL) classifier. Poria et al. experimented with both early-fusion and late-fusion methods and showed that early-fusion was more effective. Both works showed that models that used multimodal features performed better than those that only used unimodal features. In contrast, our model is trained in an end-to-end fashion, avoiding the need to train different parts separately. The model is trained to jointly extract representations from the different modalities under one loss function.

Perez-Rosas et al. [16], Jin et al. [12], and Brilman et al. [2] all extracted high-level knowledge-based features to be used in a support vector machine (SVM) classifier. Perez-Rosas et al. looked at the problem of multimodal sentiment analysis in YouTube video reviews using acoustic, visual, and bag-of-words textual features to find that multimodal systems outperform unimodal ones. Jin et al. used OpenSmile [8] and bag-of-words features to recognize emotions and compare the early- and late-fusion methods to find that late-fusion performs best. Finally, Brilman et al. extracted a comprehensive set of multimodal features, and then performed an analysis to identify features that are most indicative of successful debate performance. Brilman et al. showed that the audio modality was most predictive and a multimodal system, via late-fusion, outperforms unimodal systems.

In contrast, in our work we do not rely on high-level features, the development of which requires expert-knowledge in speech and language processing. In addition, we consider neural approaches to multimodal modelings instead of SVM-based ones. The inputs to our model consist of frequency-domain representation of speech signals and `word2vec` feature representations. In our work we also investigate different pooling strategies and their impact on overall performance.

## 3 DATASET AND FEATURES

**Dataset.** We focus our study on the IEMOCAP dataset [3]. IEMOCAP was collected to elicit realistic dyadic interactions between actors. Each utterance in IEMOCAP was labeled for both valence and arousal on a 5-point Likert scale by at least two distinct annotators. We use the 10,032 utterances that have both the acoustic and lexical content. An utterance in IEMOCAP has an average duration of 4.5 seconds with a standard deviation of 1.9 seconds. We chose to use IEMOCAP because: (1) it is one of the largest emotion datasets; (2) it provides both the `.wav` files and their associated transcripts; (3) all utterances were recorded in English.

**Labels.** We convert the 5-point scale used for describing valence values to a 3-point scale following the approach described by Chang et al. [5]. This is done by pooling valence levels 1 and 2 into a single "low" value and pooling valence levels 4 and 5 into a single "high" value. We generate fuzzy labels for each utterance by representing the labels from each annotator as one-hot vectors and computing the mean over the vectors. For instance, if three annotators labeled an utterance [0, 0, 1], [0, 0, 1], and [0, 1, 0] each, then the final label vector representation would be [0, 0.3, 0.7] and the correct class label would be 2 (where the possible options are {0, 1, 2}). We treat the problem as a three-way classification problem, where the goal is to assign a label from {0, 1, 2} to a given utterance.

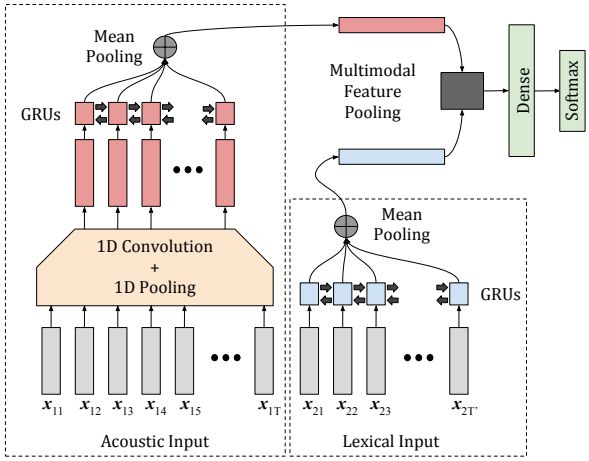


**Figure 1: Overall network architecture.**

**Acoustic Features.** We extract 40 Mel Filterbank (MFB) features by sliding a 25 millisecond Hamming window with a step-size of 10 milliseconds. As a result, each utterance is represented as a sequence of 40-dimensional feature vectors. MFBs have shown success in many speech processing applications, including speech recognition and emotion recognition [4, 21].

**Lexical Features.** We represent each word in the dataset as a 300-dimensional vector using a pre-trained `word2vec` model[1]. `word2vec` representations have shown success in sentiment analysis tasks [13]. Sentiment analysis from text is closely related to predicting valence in emotional speech. Thus, we expect `word2vec` representations to be useful for our task.

## 4 METHODS

### 4.1 Architecture

The multimodal architecture that we use is shown in Figure 1. The hyper-parameters that we consider are shown in Table 1. The network architecture accepts two input streams, one for each modality. The acoustic input stream takes a sequence of 40-dimensional vectors, while the lexical input stream takes a sequence of 300-dimensional vectors.

**Acoustic Input.** We pass the sequence of acoustic features through five layers of 1D convolution and 1D max-pooling to reduce the temporal resolution of the acoustic input sequence by $2^5$, in order to make training faster (since the acoustic features have a temporal resolution of 10 milliseconds). We then pass the resulting sequence to bidirectional gated recurrent unit (GRU) layers [7] for temporal modeling. Previous work showed that GRUs can have comparable performance to that of long short-term memory (LSTM)

[1] https://code.google.com/archive/p/word2vec/

**Table 1: Hyper-parameters used in the validation process.**

| Hyper-parameter | Values |
|---|---|
| number of conv. kernels | {64, 128} |
| conv. kernel width | {2} |
| number of conv. layers | {5} |
| 1D max-pooling kernel width | {2} |
| number of GRU layers | {1, 2} |
| GRU layers width | {32, 64} |
| number of dense layers | {0, 1} |
| dense layers width | {0, 128} |
| CBP output width | {256, 1024, 2048} |

units while using fewer parameters [7]. One of the main differences between a GRU and an LSTM unit is that a GRU has only two gates (as opposed to three) and it does not contain internal memory cells. Given the output sequence representation from the GRU layers, we induce a fixed-length feature vector by averaging the sequential outputs as described in [11], since it was shown that this can result in better discrimination between emotions when compared to only considering the output of the last layer.

**Lexical Input.** We pass the sequence of lexical feature vectors through bidirectional GRU layers and then induce a fixed-length representation by taking the average as we did for the acoustic features. We do not pass the lexical features through initial convolution or pooling because sequences of lexical features are much shorter than those of acoustic features.

**Multimodal Pooling.** For the unimodal systems, we feed the output from the average pooling layer to fully-connected layers before feeding them into a softmax layer (i.e., we skip the multimodal feature pooling step in Figure 1). For the multimodal systems, we pool the features obtained from the two modalities using the strategies described below and then feed the resulting features into fully-connected layers followed by a softmax layer.

## 4.2 Pooling Strategies

Given the representations for each modality, the next step is to pool these two representations to form a shared multimodal representation to be used for further modeling and prediction. We consider the following pooling strategies to combine the lexical and acoustic intermediate representations: (1) concatenation; (2) element-wise addition; (3) element-wise multiplication; (4) outer-product; (5) compact bilinear pooling (CBP). Unlike traditional pooling methods, outer-product and CBP provide a more expressive way to consider the interactions between features from the two modalities. Taking the outer-product of two feature vectors considers the interactions between each pair of features from the two vectors. The problem with taking the outer-product, however, is the quadratic increase in the number of parameters. CBP [10] can be used to compress the results obtained from an outer-product. In particular, we utilize the multimodal variant of CBP [9], which makes taking the outer-product between multimodal vectors more feasible.

## 4.3 CBP

Given two input vectors, $x$ and $y$, bilinear pooling is simply a linear transformation that considers all pairs of features from the two input vectors. Bilinear pooling can be obtained by first taking the outer-product of the two input vectors, $(x \otimes y)$, and then following it by a dense layer. CBP can be thought of as a sampling based approximation to bilinear pooling. The approximation is done using Tensor Sketch Projection [17, 18], and utilizes the property that $\Psi(x \otimes y, h, s) = \Psi(x, h, s) * \Psi(y, h, s)$, where $\Psi$ is the projection function, $h$ and $s$ are vectors of randomly sampled parameters, and $*$ is the convolution operation. This property obviates the need for computing outer-products of the two input vectors directly. The projection function is computed as follows: $\Psi(x, h, s)_i = \sum_{j:h_j=i}(s_j \cdot v_j)$, where $x, h, s \in \mathbb{R}^n$, $h_j$ is sampled from $\{1, ..., d\}$, $s_j$ is sampled from $\{-1, 1\}$, and $d$ is the desired output dimension. In this work we use the CBP implementation by Ronghang Hu[2].

**Table 2: Performance obtained using different pooling strategies. We assert significance when $p < 0.05$ under a paired t-test.**

| Method | UAR | $\rho$ |
|---|---|---|
| unimodal—acoustic | .590 | .320 |
| unimodal—lexical | .648$^{\ddagger}$ | .540$^{\ddagger}$ |
| concatenation | .680$^{\dagger}$ | .581$^{\dagger}$ |
| summation | .683$^{\dagger}$ | .578$^{\dagger}$ |
| multiplication | .687$^{\dagger}$ | .588$^{\dagger}$ |
| outer-product | **.694**$^{\dagger}$ | .601$^{\dagger *}$ |
| CBP | .693$^{\dagger}$ | **.605**$^{\dagger *}$ |

‡: significantly better than unimodal—acoustic
†: significantly better than unimodal—lexical and —acoustic
*: significantly better than concatenation

# 5 EXPERIMENTS

## 5.1 Recipe

We follow a leave-one-speaker-out evaluation scheme. The dataset contains a total of five sessions, where each session has data from a male and a female speaker. This results in 10 unique speakers in total. For each fold, we use one speaker for testing and the other speaker within the same session for validation and early stopping. We use the remaining eight speakers for training.

We use unweighted average recall (UAR) and Pearson correlation ($\rho$) as our evaluation metrics. UAR is a popular metric used when dealing with imbalanced classes [20]. In cases where ground-truth labels have a tie, we accept predictions for either position as a correct answer. So if $[0, 0.5, 0.5]$ is the ground-truth label, then class labels 1 and 2 are considered correct predictions in the evaluation process. To compute Pearson correlation, we convert the network's output to numerical values by taking the expected value, similar to [5].

We implement our models using Keras [6] with a TensorFlow back-end [1]. We use RMSprop [23] to train our models and use a weighted cross-entropy loss function to account for class imbalance. We use fuzzy labels in the training process similar to [5]. We run each experiment three times to account for random initialization of the parameters and report the ensemble performance. We sweep through hyper-parameters values shown in Table 1 and pick the

[2] https://github.com/ronghanghu/tensorflow_compact_bilinear_pooling

**Table 3: Confusion matrices comparison. Columns represent predictions while rows represent ground-truth.**

**(a) lexical modality**

| | neg | neu | pos |
|---|---|---|---|
| neg | **.607** | .140 | .253 |
| neu | .233 | **.572** | .194 |
| pos | .128 | .115 | **.757** |

**(b) acoustic modality**

| | neg | neu | pos |
|---|---|---|---|
| neg | **.509** | .181 | .311 |
| neu | .239 | **.621** | .140 |
| pos | .198 | .164 | **.638** |

**(c) CBP multimodel**

| | neg | neu | pos |
|---|---|---|---|
| neg | **.705** | .144 | .151 |
| neu | .247 | **.648** | .104 |
| pos | .148 | .128 | **.724** |

combination that maximizes the validation performance for each fold. We use an initial learning rate of 0.001. Starting from epoch five, we reduce the learning rate by half whenever the validation UAR does not improve at the end of each epoch.

## 5.2 Results

Table 2 shows the results for the different pooling strategies that we considered. The results show that the lexical modality yields significantly ($p < 0.05$) better performance than the acoustic modality does in terms of both UAR and $\rho$. This suggests that lexical cues are better for predicting valence than acoustic cues. The results show that multimodal systems significantly ($p < 0.05$) outperform the unimodal lexical systems, suggesting that adding the acoustic modality can still be beneficial. Pooling through element-wise multiplication provided a non-significant improvement in performance over element-wise summation and concatenation approaches. Outer-product methods provided significant improvement ($p < 0.05$) in $\rho$ when compared to results from concatenation method. Finally, our results suggest that a CBP strategy does not provide an advantage over simple outer-product strategy. This is probably due to the relatively low dimensionality of our multimodal representations required for each modality ($32 - 64$ for each modality).

Table 3 shows the confusion matrices obtained from the two unimodal systems and the CBP model. The results in Table 3 suggest that the acoustic modality is better for predicting neutral valence than the lexical modality. On the other hand, our results suggest that the lexical modality is better for predicting positive/negative valence than the acoustic modality. Finally, Table 3 shows that the significantly improved performance of CBP over that of the unimodal systems is due to more accurate negative and neutral valence predictions.

## 5.3 Analysis

The model that we use in this work abstracts the influence of individual modalities on the final decision. To further analyze the influence of each modality on the overall performance of our multimodal system, we study the effect of perturbing the individual input streams by adding white Gaussian noise (with zero mean and varying standard deviation) to the input features with different signal-to-noise-ratio (SNR) levels. We run this analysis on our best performing system, the CBP multimodal system, and vary the SNR levels from -18 dB to 6 dB. We also include SNR values of -Inf dB and +Inf dB in our analysis. The idea is that if an input modality is less important, then perturbing its values with noise will have minimal effect on the overall performance.

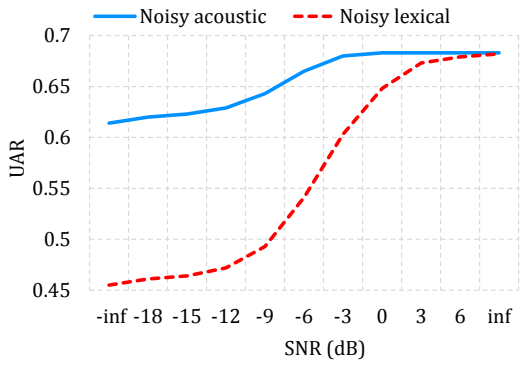


**Figure 2: Effect of adding noise to each modality (while keeping the other modality clean) on the performance of CBP multimodal system.**

Figure 2 shows the results that we obtain for this analysis. The figure shows that adding more noise to the lexical modality (dashed line) results in a rapid drop in performance compared to the performance drop due to adding noise to the acoustic modality (solid line). This suggests that the lexical modality has larger influence on the overall performance of the system. The figure shows that a multimodal system would still result in > 60% UAR even when SNR is zero for the acoustic modality.

# 6 CONCLUSION

There are several strategies that can be used to pool representations learned for acoustic and lexical modalities in neural networks. In this paper, we presented a comparison between different multimodal feature pooling strategies for the task of predicting valence in emotional speech. Our results on the IEMOCAP dataset suggest the following: (1) multimodal methods that combine acoustic and lexical features are better than unimodal for predicting valence; (2) lexical modality is better for predicting valence than the acoustic modality; (3) outer-product-based pooling strategies outperform other pooling techniques.

For future work, we plan to study pooling strategies that can be applied to temporal signals directly before inducing fixed-length representations. Such strategies should allow modeling temporal dependencies between the two signals.

# ACKNOWLEDGMENTS

This work was partially supported by IBM under the Sapphire project and by the National Science Foundation (NSF CAREER 1651740).

## REFERENCES


[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, and others. 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467* (2016).

[2] Maarten Brilman and Stefan Scherer. 2015. A multimodal predictive model of successful debaters or how i learned to sway votes. In *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 149–158.

[3] Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N Chang, Sungbok Lee, and Shrikanth S Narayanan. 2008. IEMOCAP: Interactive emotional dyadic motion capture database. *Language resources and evaluation* 42, 4 (2008), 335.

[4] Carlos Busso, Sungbok Lee, and Shrikanth S Narayanan. 2007. Using neutral speech models for emotional speech analysis.. In *Interspeech*. 2225–2228.

[5] Jonathan Chang and Stefan Scherer. 2017. Learning representations of emotional speech with deep convolutional generative adversarial networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE.

[6] François Chollet and others. 2015. Keras. https://github.com/fchollet/keras. (2015).

[7] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555* (2014).

[8] Florian Eyben, Martin Wöllmer, and Björn Schuller. 2010. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM international conference on Multimedia*. ACM, 1459–1462.

[9] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. 2016. Multimodal compact bilinear pooling for visual question answering and visual grounding. *arXiv preprint arXiv:1606.01847* (2016).

[10] Yang Gao, Oscar Beijbom, Ning Zhang, and Trevor Darrell. 2016. Compact bilinear pooling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 317–326.

[11] Sayan Ghosh, Eugene Laksana, Louis-Philippe Morency, and Stefan Scherer. 2015. Learning representations of affect from speech. *arXiv preprint arXiv:1511.04747* (2015).

[12] Qin Jin, Chengxin Li, Shizhe Chen, and Huimin Wu. 2015. Speech emotion recognition with acoustic and lexical features. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 4749–4753.

[13] Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* (2014).

[14] Linchuan Li, Zhiyong Wu, Mingxing Xu, Helen Meng, and Lianhong Cai. 2016. Combining CNN and BLSTM to Extract Textual and Acoustic Features for Recognizing Stances in Mandarin Ideological Debate Competition. *Interspeech 2016* (2016), 1392–1396.

[15] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. 2015. Bilinear cnn models for fine-grained visual recognition. In *Proceedings of the IEEE International Conference on Computer Vision*. 1449–1457.

[16] Verónica Pérez-Rosas, Rada Mihalcea, and Louis-Philippe Morency. 2013. Utterance-Level Multimodal Sentiment Analysis.. In *ACL (1)*. 973–982.

[17] Ninh Pham and Rasmus Pagh. 2013. Fast and scalable polynomial kernels via explicit feature maps. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 239–247.

[18] Hamed Pirsiavash, Deva Ramanan, and Charless C Fowlkes. 2009. Bilinear classifiers for visual recognition. In *Advances in neural information processing systems*. 1482–1490.

[19] Soujanya Poria, Erik Cambria, and Alexander F Gelbukh. 2015. Deep Convolutional Neural Network Textual Features and Multiple Kernel Learning for Utterance-level Multimodal Sentiment Analysis.. In *EMNLP*. 2539–2544.

[20] Andrew Rosenberg. 2012. Classifying Skewed Data: Importance Weighting to Optimize Average Recall.. In *Interspeech*. 2242–2245.

[21] Tara N Sainath, Brian Kingsbury, Abdel-rahman Mohamed, and Bhuvana Ramabhadran. 2013. Learning filter banks within a deep neural network framework. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 297–302.

[22] BjöRn Schuller, Stefan Steidl, Anton Batliner, Felix Burkhardt, Laurence Devillers, Christian MüLler, and Shrikanth Narayanan. 2013. Paralinguistics in speech and languageâĂŤState-of-the-art and the challenge. *Computer Speech & Language* 27, 1 (2013), 4–39.

[23] Tijmen Tieleman and Geoffrey Hinton. 2012. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning* 4, 2 (2012).