DDDM: A Brain-Inspired Framework for Robust Classification

Xiyuan Chen[1,2*], **Xingyu Li**[3*], **Yi Zhou**[4,3†] and **Tianming Yang**[1,3†]
[1]Institute of Neuroscience, Center for Excellence in Brain Science and Intelligence Technology, Chinese Academy of Sciences, Shanghai, China
[2]University of Chinese Academy of Sciences, Beijing, China
[3]Shanghai Center for Brain Science and Brain-Inspired Technology, Shanghai, China
[4]National Engineering Laboratory for Brain-inspired Intelligence Technology and Application, School of Information Science and Technology, University of Science and Technology of China, Hefei, China
{chenxiyuan, tyang}@ion.ac.cn, xyli1905@bsbii.cn, yizhoujoey@gmail.com

Abstract

Despite their outstanding performance in a broad spectrum of real-world tasks, deep artificial neural networks are sensitive to input noises, particularly adversarial perturbations. On the contrary, human and animal brains are much less vulnerable. In contrast to the one-shot inference performed by most deep neural networks, the brain often solves decision-making with an evidence accumulation mechanism that may trade time for accuracy when facing noisy inputs. The mechanism is well described by the Drift-Diffusion Model (DDM). In the DDM, decision-making is modeled as a process in which noisy evidence is accumulated toward a threshold. Drawing inspiration from the DDM, we propose the Dropout-based Drift-Diffusion Model (DDDM) that combines test-phase dropout and the DDM for improving the robustness for arbitrary neural networks. The dropouts create temporally uncorrelated noises in the network that counter perturbations, while the evidence accumulation mechanism guarantees a reasonable decision accuracy. Neural networks enhanced with the DDDM tested in image, speech, and text classification tasks all significantly outperform their native counterparts, demonstrating the DDDM as a task-agnostic defense against adversarial attacks.

1 Introduction

Deep learning has shown outstanding performance in many areas, including image, audio , and text classification. Nevertheless, deep learning lacks robustness; that is, deep artificial neural networks are susceptible to manipulations of inputs. Small perturbations on inputs that are imperceptible to humans may result in significant differences in outputs of deep neural networks [Goodfellow *et al.*, 2015]. Such susceptibility is not only of theoretical interest but also represents a severe practical problem, which causes security issues in real-world applications, including face recognition and autonomous driving.

Recently, a huge amount of effort has been dedicated to addressing this issue. Adversarial training [Madry *et al.*, 2019] is considered to be the most promising. It adds adversarial samples to the training set and automatically lets the network model adapt to them through learning. Despite these efforts, current defenses against adversarial attacks are still far from satisfactory. Most defenses only target one or a few types of attacks and are often restricted to particular tasks, such as image classification. Consequently, it is unclear whether they can be applied to other kinds of attacks or tasks.

In contrast, humans and animals are much less vulnerable. Even though time-limited human subjects can be misled by deliberately crafted images, they successfully restore accuracy by spending more time facing these uncertain and noisy inputs [Elsayed *et al.*, 2018]. The more uncertain the input is, the longer the brain takes to make a decision. This ability of trading decision speed for accuracy has been well studied in humans and animals by researchers in neuroscience and psychology [Ratcliff and Rouder, 1998; Roitman and Shadlen, 2002]. In particular, the celebrated Drift-Diffusion Model (DDM) and its variants stand as one of the most successful models that account for the decision-making process in humans and animals. The DDM describes a sequential sampling process in which signals with both noises from the external sources and the internal neuronal noises are accumulated as evidence for decisions. The decision is made once the accumulated evidence reaches a predetermined threshold. A high threshold leads to accurate but slow decisions, while a low threshold results in fast but inaccurate decisions. Neuronal activities in decision-making-related brain areas exhibit response patterns described by the DDM [Roitman and Shadlen, 2002].

The ability to trade speed for accuracy characterizes one of the most striking differences between biological and artificial neural networks. Enhancing artificial neural networks with a dynamic inference process similar to those in human and animal brains would potentially improve their robust-

*Co-first authors
†Corresponding authors

[1]The full version of our paper: https://github.com/zhouyilab/DDDM/blob/master/DDDM-IJCAI22.pdf; Implementation: https://github.com/XiYuan68/DDDM.

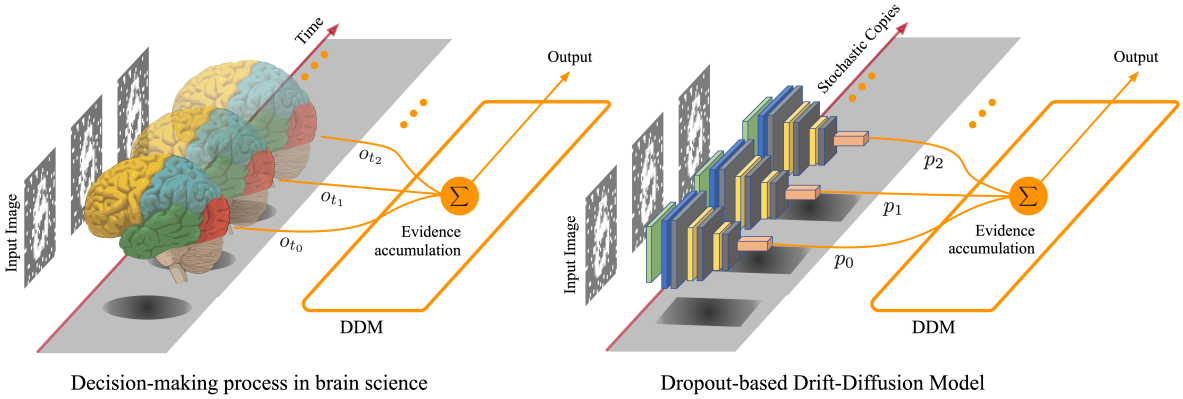

Figure 1: Schematic diagram of the Dropout-based Drift-Diffusion Model (right) and its brain science counterpart (left). In DDDM, outputs p_i from the stochastic copies of an arbitrary neural network simulate the noisy temporal neural signal o_{t_i} in the brain. In both, the series of outputs/signals are passed to a similar evidence accumulation mechanism (as described by the DDM) for a robust output.

ness. Guided by this intuition, we propose the Dropout-based Drift-Diffusion Model (DDDM) as a general framework for enhancing the robustness of arbitrary neural networks. Depicted in Figure 1, in contrast to adversarial training, which adds noises to inputs during the training phase, DDDM adds noises to neural networks via random dropouts during the test phase, resulting in multiple stochastic copies of the original one. Then, DDDM employs a DDM-like evidence accumulation mechanism to make the decision based on these outputs.

We conduct experiments on three types of datasets, including MNIST [Lecun *et al.*, 1998] and CIFAR10 [Krizhevsky, 2009] for image classification, the SpeechCommands [Warden, 2018] for audio classification, and the IMDB dataset [Maas *et al.*, 2011] for text classification. We compare the networks' performance with and without the DDDM against a variety of adversarial attacks, including both the white-box and the black-box ones. The experimental results show that the DDDM improves the robustness of the network, providing defense against all adversarial attacks tested on all three datasets.

2 Related Work

Adversarial training (AT) [Goodfellow *et al.*, 2015; Madry *et al.*, 2019] is one of the most popular adversarial defense methods. It augments the clean data with adversarial ones during the trianing process. There are works that try to improve robustness by introducing additional noise in their models. For example, [Guo *et al.*, 2018] applied transformation on images before feeding the network, including image quilting, total variance minimization, JPEG compression, and bit-depth reduction. Similar to our work, [Dhillon *et al.*, 2018] randomly dropped neurons with a weighted distribution. Note that all the defense methods mentioned above focus on either filtering out adversarial perturbations or creating obfuscated gradients [Athalye *et al.*, 2018], leaving room for more sophisticated attacks. On the contrary, our framework injects randomness into the network model during the testing phase, and the resulting stochastic predictions are agnostic to the type of adversarial attacks. Furthermore, a temporal evidence accumulation mechanism is employed to offset the accuracy reduction due to the added randomness and retain a good performance with robustness.

In neuroscience, recent advances in the study of perceptual decision-making have started to reveal the underlying neural mechanism [Gold and Shadlen, 2007]. The random-dot motion (RDM) direction discrimination task is one of the most popular tasks for studying decision-making with noisy evidence [Roitman and Shadlen, 2002]. In this task, subjects are asked to judge whether a patch of noisy dots moves toward left or right. The subjects' decision accuracies and response times can be well described by the DDM [Ratcliff and McKoon, 2008]. Furthermore, it was demonstrated that neurons in the posterior parietal cortex increased their firing when evidence toward their favored choice was being accumulated [Roitman and Shadlen, 2002]. The response increase correlated with the quality of evidence, which may be quantified as the log-likelihood ratio [Gold and Shadlen, 2007; Yang and Shadlen, 2007], and reflected the evidence accumulation process. Additional studies extended DDM into probabilistic inferences based on images and identified a prefrontal-parietal circuitry that carries out evidence accumulation process in probabilistic inferences [Yang and Shadlen, 2007]. Response patterns that reflect evidence accumulation were also found in many other brain regions, including multiple regions in the prefrontal cortex and the basal ganglia, suggesting evidence accumulation is a universal decision-making mechanism employed by the brain [Gold and Shadlen, 2007]. DDM and the related sequential sampling models in neuroscience describe the brain's solutions for optimizing decision-making with noisy inputs. However, the DDM has not been used together with deep neural networks in the machine learning field and has not been considered as a defense mechanism against adversarial attacks.

3 Model Description

3.1 Test-Phase Dropout

Dropout was initially proposed by [Hinton *et al.*, 2012] to address the problem of overfitting. In this paper, we use it to

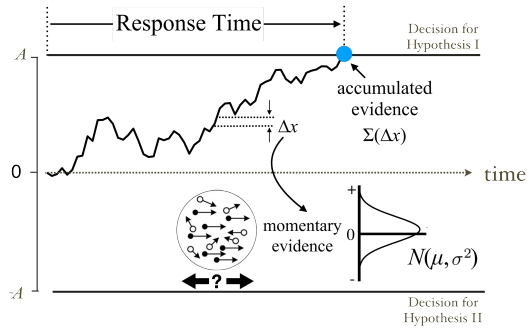

Figure 2: A diagram of the drift-diffusion model (DDM). At each moment, the noisy momentary evidence is accumulated until it hits one of the two decision thresholds (A and $-A$). The decision threshold and the mean and the variance of the momentary evidence together determine the speed and accuracy of the decision.

introduce randomness in the system. It simulates the internal neuronal noise in synaptic transmissions [Rusakov *et al.*, 2020]. With dropouts applied temporally, the original network model effectively extends to an ensemble of stochastic copies, whose outputs serves as the evidence. Let $h_{a,b}$ be an arbitrary deep neural network with one or several dropout layers. The dropout layers share the same dropout rate. Subscript a denotes the dropout rate used for model training and validation, and b denotes the dropout rate at the test phase. We refer to $h_{a,b}$ as the dropout classifier in the following context to emphasize the involvement of the test-phase dropout mechanism.

3.2 Drift-Diffusion Model

In a typical two-choice scenario, the canonical Drift-Diffusion Model (DDM) (see Figure 2) describes decision-making as a process in which evidence that is noisy and fluctuates over time (momentary evidence) is accumulated toward a preset threshold A. Mathematically, DDM corresponds to the following stochastic differential equation

$$\tau \dot{X} = \mu + \xi(t), \tag{1}$$

where X is known as the decision variable representing the cumulated evidence, and μ is the drift or the mean of the noisy momentary evidence. The Gaussian process $\xi(t)$, with mean 0 and variance σ^2, captures the uncertainty in the evidence. Coefficient τ is the characteristic time scale. The decision variable X evolves until reaching one of the two thresholds $\pm A$ at which the corresponding decision is made. These thresholds correspond to the choice options, and the time needed for the accumulated evidence to reach a particular threshold reflects the response time (RT).

The canonical DDM can be extended to decisions with more than two alternatives. Equation (1) can be derived from a set of stochastic differential equations [Roxin, 2019]. Following this route, it is straightforward to increase the number of choices by extending the number of differential equations. Another route to extend the DDM to the multi-choice domain models the decision-making as a Bayesian multiple sequential probability ratio test (MSPRT). It has been demonstrated the two routes are equivalent [Roxin, 2019].

For our purpose to integrate the DDM with the test-phase dropout classifier, we use the Bayesian MSPRT route. Suppose there are n alternatives, and $z_i(t)$ denotes the instantaneous evidence for choice i. $z_i(t)$ is drawn from a Gaussian distribution with mean M_i and variance σ^2. Let $\mathcal{O}(T) := \{z_i(t) : 1 \le i \le n, 1 \le t \le T\}$ be the collection of observations up to a time T, and H_i denote the hypothesis that choice i has the largest mean. Within the Bayesian MSPRT framework, one seeks to make a decision based on the posterior $\Pr(H_i|\mathcal{O}(T))$. With defining

$$X_k(t) := \sum_i^k y_i(t) - k y_{k+1}(t), \quad 1 \le k \le n-1,$$

where $y_i(t) := \int_0^t dt'\, z_i(t')$ and X_k is the decision variable for the kth class, measuring how likely the original input belongs to class k given the accumulated evidence. It can be shown that X_k satisfies a stochastic differential equation of the same type as Equation (1). We refer the interested readers to [Roxin, 2019] for a complete deduction.

3.3 Dropout-based Drift-Diffusion Model

There are two components in the design of the Dropout-based Drift-Diffusion model (DDDM). The first component turns the static one-shot inference process into a dynamic, noisy inference process. The noisy predictions from the stochastic copies of neural networks simulate the series of temporal signals in the biological brain. Presently, we implement the dynamic inference by introducing the test-phase dropout, which will randomly disturb the loss landscape and potentially make adversarial attacks harder to succeed. This results in a dropout classifier whose outputs are subject to further processing.

For the second component, we take the noisy outputs from the test-phase dropout classifier as evidence and apply DDM with the Bayesian MSPRT implementation. More precisely, for a dropout classifier $h_{a,b}$, let C be the number of categories and x be an input. We arrange L predictions to form a trial

$$T = \begin{bmatrix} p_{11} & \cdots & p_{1C} \\ \vdots & \ddots & \vdots \\ p_{L1} & \cdots & p_{LC} \end{bmatrix},$$

where p_{ij} is the i-th predicted confidence of x belonging to label j. Let T_i denote the i-th row vector in matrix T, we compute the posterior at time step t

$$P(j,t) := \Pr(H_j|\{T_1, \ldots, T_t\}),\ 1 \le j \le C, 1 \le t \le L.$$

Let $RT \le L$ be the smallest time step at which there exists a label $winner$ satisfying $P(winner, RT) \ge A$. We say that x belongs to the label $winner$ with a response time RT. If the threshold is never surpassed during the whole trial, we will force a decision by assigning the label winner to be the one with the largest decision variable at time step L, i.e., $P(winner, L) = \max(P(1,L), P(2,L), \ldots, P(C,L))$. In practice, we apply multiple trials and an appropriate decision threshold A is chosen to offset the noise introduced by dropouts and ensure overall accuracy. To compute the posterior $P(i,j)$, we apply Bayes theorem and assume equal

	clean	FGSM	PGD	L_2 C&W	L_2 DF	Salt&Pepper	Uniform	Spatial	Square
$h_{0,0}$ (%)	98.10	12.10	0.00	1.70	42.70	0.20	94.50	0.40	19.50
$h^*_{a,b}$ (%)	99.20	50.70	43.30	96.70	67.90	96,90	97.10	34.80	78.00
$(a,b)^*$	$(0.2, 0.0)$	$(0.0, 0.6)$	$(0.0, 0.6)$	$(0.4, 0.4)$	$(0.2, 0.6)$	$(0.2, 0.4)$	$(0.2, 0.2)$	$(0.6, 0.8)$	$(0.0, 0.2)$
$H^*_{a,b}$ (%)	99.16	73.12	74.85	98.89	95.04	98.89	98.24	65.22	88.93
$(a,b)^*$	$(0.4, 0.2)$	$(0.0, 0.6)$	$(0.0, 0.8)$	$(0.2, 0.4)$	$(0.6, 0.8)$	$(0.4, 0.4)$	$(0.2, 0.4)$	$(0.4, 0.8)$	$(0.0, 0.6)$
h_{a^*,b^*} (%)	85.90	49.70	36.40	85.40	67.90	86.70	79.10	30.20	64.10
H_{a^*,b^*} (%)	98.20	71.09	51.68	97.94	92.30	98.08	97.36	36.73	87.15

Table 1: Performance of the DDDM under eight adversarial attacks on MNIST. The optimal performance of the DDDM $H^*_{a,b}$ for individual attacks are compared to the ones of the dropout classifier $h^*_{a,b}$. The accuracy of the undefended model $h_{0,0}$ is included as the baseline. For each case, we show the prediction accuracy of the individual optimal models $h^*_{a,b}/H^*_{a,b}$ as well as their corresponding optimal dropout rate pair $(a,b)^*$. We also include the overall performance of the DDDM H_{a^*,b^*} and the dropout classifier h_{a^*,b^*}. The global optimal dropout rate pair (a^*, b^*) is chosen as $(0.2, 0.6)$.

priors for all hypotheses H_i. The problem is then reduced to computing the likelihoods $\Pr(\{T_1, \ldots, T_t\}|H_i)$, which are approximately evaluated over the clean training data. The detailed description of our implementation can be found in the supplementary material.

In the following context, we will denote the dropout classifier equipped with DDM as $H_{a,b}$ and refer to it as DDDM for simplicity.

4 Experiments and Results

The primary goal of the present experiments is to evaluate the robustness of our DDDM model. Our experiments cover three different modalities of data, i.e., image, audio, and text. Under all types of inputs, the DDDM is demonstrated to be effective in defending adversarial attacks.

For all the tasks, we run experiments over different combinations of dropout rates (a, b), both taking values from the set $\{0, 0.2, 0.4, 0.6, 0.8\}$. Therefore, twenty-five classifiers are tested in each case. In implementing DDM, we draw 100 predictions from each classifier on each generated adversarial example. Then, 10 trials of the length $L = 25$ are randomly sampled from those predictions. These trials are fed into the evidence accumulation mechanism with a decision threshold $A = 0.99$ for the final prediction.

4.1 Images Classification

MNIST Dataset

In the experiments, we adopt the same network architecture as in [Madry *et al.*, 2019] with an additional dropout layer that follows each layer except the input/output ones.

To comprehensively evaluate the effectiveness of DDDM in defending adversarial attacks, we considered four white-box attacks and four black-box attacks as listed below: the Fast Gradient Sign Method (FGSM) attack [Goodfellow *et al.*, 2015], the Projected Gradient Descent (PGD) attack [Madry *et al.*, 2019], the L_2 Carlini and Wagner (L_2 C&W) attack [Carlini and Wagner, 2017], the L_2 DeepFool attack [Moosavi-Dezfooli *et al.*, 2016], the Salt and Pepper attack, the L_∞ uniform noise attack [Rauber and Bethge, 2020], the Spatial attack [Engstrom *et al.*, 2019] and the Square attack [Andriushchenko *et al.*, 2020]. The details of attack settings can be found in the supplementary material.

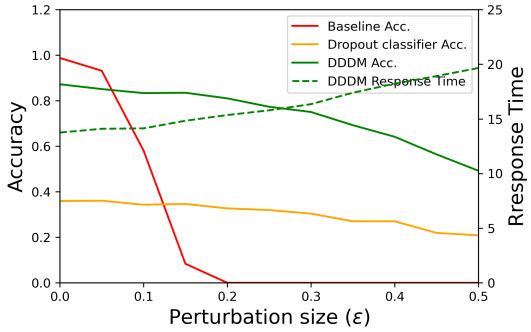

Figure 3: Variation of accuracy (solid lines) and response time (the dashed line) against increasing perturbation size (ϵ). We compared three models: the baseline model $h_{0,0}$, the dropout classifier without DDM $h_{0.0,0.8}$, and the dropout classifier with DDM $H_{0.0,0.8}$ under the PGD attack on MNIST.

In Table 1, we compare the optimal DDDMs ($H^*_{a,b}$) with the optimal dropout classifiers ($h^*_{a,b}$) on individual attacks and the clean data. The undefended model ($h_{0,0}$) is included as the baseline. To demonstrate the effectiveness of DDDM under the combination of all attacks, we also include the results for H_{a^*,b^*} and h_{a^*,b^*}, i.e., the DDDM and the dropout classifier with a single optimal dropout rate pair (a^*, b^*) across all the attacks. (a^*, b^*) is determined with the criterion that the corresponding H_{a^*,b^*} should try its best to avoid sacrificing the clean accuracy and, at the same time, provide as much defense as possible across different attacks. The results indicate that the test-phase dropouts effectively neutralize the adversarial perturbations against all attacks in our experiments, and the DDM retains the accuracy through evidence accumulation. Furthermore, H_{a^*,b^*} outperforms h_{a^*,b^*} as well as $h_{0,0}$ on all the cases.

Trade-off between Response Time and Accuracy. Animals and humans spend more effort in making difficult decisions, and the DDDM exhibits the same behavior. While the randomness in the test-phase dropout classifier neutralizes the

	clean	PGD	L_2 DF	Spatial
$h_{0,0}$ (%)	89.71	5.24	67.96	8.93
$h^*_{a,b}$ (%)	89.89	68.46	71.53	40.98
$(a,b)^*$	$(0.2, 0.0)$	$(0.8, 0.8)$	$(0.0, 0.2)$	$(0.4, 0.8)$
$H^*_{a,b}$ (%)	89.67	68.46	79.01	64.26
$(a,b)^*$	$(0.6, 0.4)$	$(0.8, 0.8)$	$(0.0, 0.8)$	$(0.0, 0.8)$
h_{a^*,b^*} (%)	86.88	35.87	67.57	36.92
H_{a^*,b^*} (%)	89.30	36.76	70.00	37.59

Table 2: Performance of the DDDM under three attacks on the CIFAR10 dataset. The same as in Table 1, we include the performance under individual attacks as well as the overall performance with the global optimal dropout rate pair $(a^*, b^*) = (0.6, 0.8)$.

adversarial perturbations, the DDM helps retain a decent accuracy by spending more time in inference. To demonstrate this trade-off, we plot in Figure 3 the accuracies and response times with respect to different adversarial perturbation sizes, comparing DDDM to the dropout classifier as well as the undefended baseline model. The response time is measured by the number of forward passes. The accuracy of the baseline model drops rapidly as the perturbation size increase. On the other hand, both the DDDM and the test-phase dropout classifier retain relatively high accuracy even when the perturbation is large. For small perturbations, the test-phase dropout classifier's accuracy curve lies well below the baseline model's curve. In contrast, the performance of DDDM dominates on almost the whole perturbation range except for near 0. Finally, the response time increases monotonically with perturbation level, demonstrating a trade for accuracy with time.

CIFAR10 Dataset

We further carry out experiments on the CIFAR10 [Krizhevsky, 2009] dataset to challenge our approach under a more realistic scenario. To implement the test-phase dropout classifier, we use the VGG16 architecture [Simonyan and Zisserman, 2015] without batch normalization. A dropout layer is added to each of the last six convolutional layers. We will discuss more the method of adding dropout layers in the supplementary material.

The results are summarized in Table 2, which includes three effective attacks in our test: PGD, L2DF, and Spatial. Note that these attacks cover both white-box and black-box attack categories. Again, DDDM effectively defends all the attacks with only a minor sacrifice on the clean accuracy. Compared to the MNIST case, while H_{a^*,b^*} keeps outperforming h_{a^*,b^*}, their performance gaps become much smaller. This may be attributed to the fact that under a large model such as VGG16, six dropout layers do not introduce sufficient randomness in the dropout classifier. Hence, DDM does not work at its full power.

4.2 Audio Classification

To evaluate the cross-domain performance of DDDM, we consider the audio classification task on SpeechCommands [Warden, 2018] dataset, which contains 35 keywords and 105829 audio clips of those keywords. In the audio domain, present works focus on the classification and recognition of audio clips, which serve as the base of various applications such as voice input and voice assistants. The existing audio adversarial attacks [Carlini and Wagner, 2018; Qin *et al.*, 2019] severely endangers the safety of such automatic speech recognition (ASR) systems.

	clean	imperceptible
$h_{0,0}$ (%)	86.73	5.18
h_{a^*,b^*} (%)	82.91	68.36
H_{a^*,b^*} (%)	85.02	70.62

Table 3: Performance of the DDDM. Similar to Table 1 but for the imperceptible attack on the SpeechCommands dataset. Since there is only one attack, we only includes the overall performance with the global optimal dropout rate pair $(a^*, b^*) = (0.4, 0.4)$.

	clean	TextBugger
$h_{0,0}$ (%)	88.44	1.7
h_{a^*,b^*} (%)	89.16	22.10
H_{a^*,b^*} (%)	89.41	86.75

Table 4: Performance of the DDDM. Similar to Table 3 but for the TextBugger attack on the IMDB dataset. The global optimal dropout rate pair is $(a^*, b^*) = (0.4, 0.4)$.

In our test, we adopt a simplified version of the DeepSpeech2 model [Amodei *et al.*, 2015] against the attack proposed by [Qin *et al.*, 2019], which adds human-imperceptible perturbations to the audio waveform to mislead the model predictions. Our DeepSpeech2 model includes a Mel-spectrogram conversion layer, followed by one 1D convolutional layer, two LSTMs, and two fully-connected layers. Dropout is only applied once after the convolutional layer. After training, we attacked the model with the adversarial perturbation bound of 0.05.

Results in Table 3 show that DDDM providea defense against the imperceptible attack with only a minor decrease in clean accuracy. Comparing the dropout classifier h_{a^*,b^*} with the DDDM H_{a^*,b^*}, the latter show better performance under clean and adversarial case situations. We notice that the gap between h_{a^*,b^*} and H_{a^*,b^*} is small. This is similar to the case for CIFAR10 and could be attributed to the insufficient number of dropout layers.

4.3 Text Classification

At last, we evaluate our approach on the text classification task. Natural languages consist of discrete characters and words. The process of mapping tokens to embeddings prevents directly applying attack methods from the image domain on textual classifiers. A text-specific attack methods have been developed [Li *et al.*, 2019].

The DDDM is evaluated on the IMDB dataset [Maas *et al.*, 2011] against the TextBugger attack [Li *et al.*, 2019]. We use an LSTM classifier resembling the one in the TextBugger work, with adding an additional dropout layer to the pre-activation outputs of each gate in the network. Words from

the IMDB dataset are first converted into pre-trained GloVe embedding vectors of dimension 300, then fed to the LSTM classifier.

We summarize the experimental results in Table 4. DDDM H_{a^*,b^*} successfully protected the classifier with a considerable improvement in performance compared to the dropout classifier h_{a^*,b^*}. The robust accuracy of H_{a^*,b^*} under the TextBugger attack approaches its clean accuracy. Moreover, we note that the clean accuracy of H_{a^*,b^*} is higher than the one of the undefended model $h_{0,0}$. This could be explained by the fact that we use the clean training data to evaluate the likelihood. We emphasize that this does not harm the effectiveness of our approach since it does not require information about the possible attacks beforehand.

5 Discussion

The current work is merely proof of the principle that combining an evidence accumulation mechanism with deep neural networks enhances robustness and provides defense against adversarial attacks. The key to the framework's success lies in the interplay between two components of the framework. The first is the mechanism to introduce noise into the system. Partly because the random noise itself degrades the performance; therefore, achieving a balance between performance and robustness poses a challenge. The second component, the evidence accumulation in the DDM, provides a mechanism to solve this problem and allows more flexibility in terms of the level of additional noise added into the system. In addition to artificially introduced noise, noise exists in many real-world situations, such as the automatic driving systems. In these cases, noise naturally presented in the input may be combined with artificially introduced noise for defense against attacks.

The attractiveness of the framework also lies in its simplicity. DDDM does not require the networks to be trained differently for any particular attacks. Instead, it is agnostic to the type of attack and its particular scheme of noise. Thus, it provides protection not only for the attacks tested here but also for other attacks. The framework also does not depend on particular types of networks. It can also be used in combination with more sophisticated networks and larger datasets.

Other than what has been mentioned above, several other directions to extend the current framework can be explored in the future. In addition to the dropouts, other ways of introducing noise into the system may help to battle perturbations. A mechanism of online adjustment of decision thresholds may help improve the speed-accuracy tradeoff in a dynamic environment. Finally, the evidence accumulation mechanism is not implemented with a neural network in the current study. However, previous studies have shown that simple recurrent networks may achieve the functionality [Wang, 2002; Wong and Wang, 2006], providing an end-to-end neural network solution for the DDDM.

6 Conclusion

Inspired by the neural mechanism of decision making, we develop a novel framework (see Figure 1) and demonstrate its effectiveness in improving the robustness of deep learning networks against adversarial attacks in different domains with a brain-like mechanism (see Tables 1 to 4). In contrast to adversarial training, the DDDM shows an advantage of sustaining the same level of clean accuracy after being attacked. By extending the one-shot classification into a temporal decision-making process based on evidence accumulation, our framework trades inference time for accuracy to achieve robustness.

To the best of our knowledge, this is the first framework to combine deep neural networks and a brain-inspired decision-making mechanism for defense against adversarial attacks and the first framework to provide cross-domain protection without adversarial training. We hope that our work brings the advances of neuroscience research to the attention of the artificial intelligence community and encourages researchers to take a closer look at the most powerful yet robust classifier —— our brain.

Acknowledgments

This work was supported by grants from the National Science and Technology Innovation 2030 Project of China to Yi Zhou (2021ZD0202600) and to Tianming Yang (2021ZD0203701). This work was also supported by grants from the Chinese Academy of Sciences (XDB32070100), and the Shanghai Municipal Science and Technology (2018SHZDZX05) to Tianming Yang.

References

[Amodei *et al.*, 2015] Dario Amodei, Rishita Anubhai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Jingdong Chen, Mike Chrzanowski, Adam Coates, Greg Diamos, Erich Elsen, Jesse Engel, Linxi Fan, Christopher Fougner, Tony Han, Awni Hannun, Billy Jun, Patrick LeGresley, Libby Lin, Sharan Narang, Andrew Ng, Sherjil Ozair, Ryan Prenger, Jonathan Raiman, Sanjeev Satheesh, David Seetapun, Shubho Sengupta, Yi Wang, Zhiqian Wang, Chong Wang, Bo Xiao, Dani Yogatama, Jun Zhan, and Zhenyao Zhu. Deep Speech 2: End-to-End Speech Recognition in English and Mandarin. *arXiv:1512.02595 [cs]*, December 2015.

[Andriushchenko *et al.*, 2020] Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square Attack: A Query-Efficient Black-Box Adversarial Attack via Random Search. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, volume 12368, pages 484–501. Springer International Publishing, Cham, 2020.

[Athalye *et al.*, 2018] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples. *arXiv:1802.00420 [cs]*, July 2018.

[Carlini and Wagner, 2017] Nicholas Carlini and David Wagner. Towards Evaluating the Robustness of Neural Networks. *arXiv:1608.04644 [cs]*, March 2017.

[Carlini and Wagner, 2018] Nicholas Carlini and David Wagner. Audio Adversarial Examples: Targeted Attacks on Speech-to-Text. January 2018.

[Dhillon *et al.*, 2018] Guneet S. Dhillon, Kamyar Azizzadenesheli, Zachary C. Lipton, Jeremy Bernstein, Jean Kossaifi, Aran Khanna, and Anima Anandkumar. Stochastic Activation Pruning for Robust Adversarial Defense. *arXiv:1803.01442 [cs, stat]*, March 2018.

[Elsayed *et al.*, 2018] Gamaleldin F. Elsayed, Shreya Shankar, Brian Cheung, Nicolas Papernot, Alexey Kurakin, Ian Goodfellow, and Jascha Sohl-Dickstein. Adversarial examples that fool both computer vision and time-limited humans. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pages 3914–3924, Red Hook, NY, USA, December 2018. Curran Associates Inc.

[Engstrom *et al.*, 2019] Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. Exploring the Landscape of Spatial Robustness. In *International Conference on Machine Learning*, pages 1802–1811. PMLR, May 2019.

[Gold and Shadlen, 2007] Joshua I. Gold and Michael N. Shadlen. The Neural Basis of Decision Making. *Annual Review of Neuroscience*, 30(1):535–574, July 2007.

[Goodfellow *et al.*, 2015] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. *arXiv:1412.6572 [cs, stat]*, March 2015.

[Guo *et al.*, 2018] Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens van der Maaten. Countering Adversarial Images using Input Transformations. In *International Conference on Learning Representations*, February 2018.

[Hinton *et al.*, 2012] Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv:1207.0580 [cs]*, July 2012.

[Krizhevsky, 2009] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, April 2009.

[Lecun *et al.*, 1998] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, November 1998.

[Li *et al.*, 2019] Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. TextBugger: Generating Adversarial Text Against Real-world Applications. *Proceedings 2019 Network and Distributed System Security Symposium*, 2019.

[Maas *et al.*, 2011] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning Word Vectors for Sentiment Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[Madry *et al.*, 2019] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards Deep Learning Models Resistant to Adversarial Attacks. *arXiv:1706.06083 [cs, stat]*, September 2019.

[Moosavi-Dezfooli *et al.*, 2016] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. DeepFool: A Simple and Accurate Method to Fool Deep Neural Networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2574–2582, June 2016.

[Qin *et al.*, 2019] Yao Qin, Nicholas Carlini, Ian Goodfellow, Garrison Cottrell, and Colin Raffel. Imperceptible, Robust, and Targeted Adversarial Examples for Automatic Speech Recognition. March 2019.

[Ratcliff and McKoon, 2008] Roger Ratcliff and Gail McKoon. The Diffusion Decision Model: Theory and Data for Two-Choice Decision Tasks. *Neural Computation*, 20(4):873–922, April 2008.

[Ratcliff and Rouder, 1998] Roger Ratcliff and Jeffrey N. Rouder. Modeling Response Times for Two-Choice Decisions. *Psychological Science*, 9(5):347–356, September 1998.

[Rauber and Bethge, 2020] Jonas Rauber and Matthias Bethge. Fast Differentiable Clipping-Aware Normalization and Rescaling. *arXiv:2007.07677 [cs, stat]*, July 2020.

[Roitman and Shadlen, 2002] Jamie D. Roitman and Michael N. Shadlen. Response of Neurons in the Lateral Intraparietal Area during a Combined Visual Discrimination Reaction Time Task. *The Journal of Neuroscience*, 22(21):9475–9489, November 2002.

[Roxin, 2019] Alex Roxin. Drift–diffusion models for multiple-alternative forced-choice decision making. *The Journal of Mathematical Neuroscience*, 9(1):5, December 2019.

[Rusakov *et al.*, 2020] Dmitri A. Rusakov, Leonid P. Savtchenko, and Peter E. Latham. Noisy Synaptic Conductance: Bug or a Feature? *Trends in Neurosciences*, 43(6):363–372, June 2020.

[Simonyan and Zisserman, 2015] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv:1409.1556 [cs]*, April 2015.

[Wang, 2002] Xiao-Jing Wang. Probabilistic Decision Making by Slow Reverberation in Cortical Circuits. *Neuron*, 36(5):955–968, December 2002.

[Warden, 2018] Pete Warden. Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition. *arXiv:1804.03209 [cs]*, April 2018.

[Wong and Wang, 2006] Kong-Fatt Wong and Xiao-Jing Wang. A Recurrent Network Mechanism of Time Integration in Perceptual Decisions. *Journal of Neuroscience*, 26(4):1314–1328, January 2006.

[Yang and Shadlen, 2007] Tianming Yang and Michael N. Shadlen. Probabilistic reasoning by neurons. *Nature*, 447(7148):1075–1080, June 2007.