A Deep Learning Approach for Multimodal Deception Detection

Gangeshwar Krishnamurthy[1], Navonil Majumder[2], Soujanya Poria[3], and Erik Cambria[4]

[1] A*STAR Artificial Intelligence Initiative (A*AI),
Institute of High Performance Computing, Singapore
gangeshwark@ihpc.a-star.edu.sg
[2] Centro de Investigacin en
Computacin, IPN, Mexico
navonil@sentic.net
[3] Temasek Laboratories,
Nanyang Technological University, Singapore
sporia@ntu.edu.sg
[4] School of Computer Science and Engineering,
Nanyang Technological University, Singapore
cambria@ntu.edu.sg

Abstract. Automatic deception detection is an important task that has gained momentum in computational linguistics due to its potential applications. In this paper, we propose a simple yet tough to beat multi-modal neural model for deception detection. By combining features from different modalities such as video, audio, and text along with Micro-Expression features, we show that detecting deception in real life videos can be more accurate. Experimental results on a dataset of real-life deception videos show that our model outperforms existing techniques for deception detection with an accuracy of 96.14% and ROC-AUC of 0.9799.

1 Introduction

We face deceptive behavior in our day-to-day life. People lie to escape from a situation that seems unfavorable to them. As a consequence, some lies are innocuous but others may have severe ramifications in the society. Reports suggest that the ability of humans to detect deception without special aids is only 54%[1]. A study by DePaulo et al. [2] found that deception without any particular motivation or intention exhibited almost no detectable cues of deception. However, cues were significantly more when lies were about transgressions.

With the rise in the number of criminal cases filed every year in the US[5], it is ethically and morally important to accuse only the guilty defendant and free the innocent. Since the judgment for any case is mostly based on the hearings and evidence from the stakeholders (accused, witnesses, etc.), the judgment is

[5] http://www.uscourts.gov

most likely to go wrong if the stakeholders do not speak the truth. It is, hence, important to detect deceptive behavior accurately in order to upkeep the law and order.

Social media can be characterized as a virtual world where people interact with each other without the human feel and touch. It is easy to not reveal one's identity and/or pretend to be someone else on the social media. Cyberbullying is increasingly becoming a common problem amongst the teenagers nowadays [3]. These include spreading rumors about a person, threats, and sexual harassment. Cyberbullying adversely affects the victim and leads to a variety of emotional responses such as lowered self-esteem, increased suicidal thoughts, anger, and depression[4]. Teenagers fall prey to these attacks due to their inability to comprehend the chicanery and pretentious behavior of the attacker.

Another area where deception detection is of paramount importance is with the increased number of false stories, a.k.a Fake News, on the Internet. Recent reports suggest that the outcome of the U.S. Presidential Elections is due to the rise of online fake news. Propagandists use arguments that, while sometimes convincing, are not necessarily valid. Social media, such as Facebook and Twitter, have become the propellers for this political propaganda. Countries around the world, such as France [5], are employing methods that would prevent the spread of fake news during their elections. Though these measures might help, there is a pressing need for the computational linguistics community to devise efficient methods to fight Fake News given that humans are poor at detecting deception.

This paper is organized as follows. In section 2, we will talk about the past work in deception detection; section 3 describes our approach to solving deception detection. Section 4 explains our experimental setup. In section 5 and 6, we discuss our results and drawbacks respectively. And finally, conclude with future work in section 7.

2 Related Work

Past research in the detection of deception can be broadly classified as Verbal and Non-verbal. In verbal deception detection, the features are based on the linguistic characteristics, such as n-grams and sentence count statistics [6], of the statement by the subject under consideration. Use of more complex features such as psycholinguistic features [7] based on the Linguistic Inquiry and Word Count (LIWC) lexicon, have also been explored by [6] and shown that they are helpful in detecting deceptive behavior. Yancheva and Rudzicz studied the relation between the syntactic complexity of text and deceptive behavior [8]. 1 In non-verbal deception detection, physiological measures were the main source of signals for detecting deceptive behavior. Polygraph tests measure physiological features such as heart rate, respiration rate, skin temperature of the subject under investigation. But these tests are not reliable and often misleading as indicated by [9, 10] since judgment made by humans are often biased. Facial expressions and hand gestures were found to be very helpful in detecting deceptive nature. Ekman [11] defined micro-expressions as short involuntary expressions, which could potentially indicate deceptive behavior. Caso et al. [12] identified particular hand gesture to be important to identify the act of deception. Cohen et al. [13] found that fewer iconic hand gestures were a sign of a deceptive narration.

Previous research was focused on detecting deceit behavior under constrained environments which may not be applicable in real life surroundings. Recently, the focus has been towards experiments in real life scenarios. Towards this, Pérez-Rosas et al. [14] introduced a new multi-modal deception dataset having real-life videos of courtroom trials. They demonstrated the use of features from different modalities and the importance of each modality in detecting deception. They also evaluated the performance of humans in deception detection and compared it with their machine learning models. Wu et al. [15] have developed methods that leverage multi-modal features for detecting detection. Their method heavily emphasizes on feature engineering along with manual cropping and annotating videos for feature extraction.

In this paper, we describe our attempt to use neural models that uses features from multiple modalities for detecting deception. We believe our work is the first attempt at using neural networks for deceit classification. We show that with the right features and simple models, we can detect deceptive nature in real life trial videos more accurately.

3 Approach

3.1 Multi-Modal Feature Extraction

The first stage is to extract unimodal features from each video. We extract textual, audio and visual features as described below.

Visual Feature Extraction For extracting visual features from the videos, we use 3D-CNN [16]. 3D-CNN has achieved state-of-the-art results in object classification on tridimensional data [16]. 3D-CNN not only extracts features from each image frame, but also extracts spatiotemporal features [17] from the whole video which helps in identifying the facial expressions such as smile, fear, or stress.

The input to 3D-CNN is a video v of dimension (c, f, h, w), where c represents the number of channels and f, h, w are the number of frames, height, and width of each frames respectively. A 3D convolutional filter, f_l of dimension (f_m, c, f_d, f_h, f_w) is applied, where f_m = number of feature maps, c = number of channels, f_d = number of frames (also called depth of the filter), f_h = height of the filter, and f_w = width of the filter. This filter, f_l, produces an output, *convout* of dimension $(f_m, c, f - f_d + 1, h - f_h + 1, w - f_w + 1)$ after sliding across the input video, v. Max pooling is applied to *convout* with window size being (m_p, m_p, m_p). Subsequently, this output is fed to a dense layer of size d_f

and softmax. The activations of this dense layer is used as the visual feature representation of the input video, v.

In our experiments, we consider only RBG channel images, hence $c = 3$. We use 32 feature maps and 3D filters of size, $f_d = f_h = f_w = 5$. Hence, the dimension of the filter, f_l, is $32 \times 3 \times 5 \times 5 \times 5$. The window size, m_p, of the max pooling layer is 3. Finally, we obtain a feature vector, v_f, of dimension 300 for an input video, v.

Textual Features Extraction We use Convolutional Neural Networks (CNN) [18, 19] to extract features from the transcript of a video, v. First, we use pre-trained Word2Vec [20] model to extract the vector representations for every word in the transcript. These vectors are concatenated and fed as input vector to the CNN. We use a simple CNN with one convolutional layer and a max-pooling layer, to get our sentence representation. In our experiments, filters of size 3, 5 and 8 with 20 feature maps each is used. Window-size of 2 is employed for max-pooling over these feature maps. Subsequently, a full-connected layer with 300 neurons is used with rectified linear unit (ReLU) [21] as the activation function. The activations of this full-connected layer is used as the textual feature representation of the input video, v. Finally, we obtain a feature vector, t_f, of dimension 300 for an input text (transcript), t.

Audio Feature Extraction openSMILE [22] is an open-source toolkit used to extract high dimensional features from an audio file. In this work, we use openSMILE to extract features from the input audio. Before extracting the features, we make sure that there are no unnecessary signals in the audio that affects the quality of the extracted features. Hence, the background noise is removed from the audio and Z-standardization is used to perform voice normalization. To remove the background noise, we use SoX (Sound eXchange) [23] audio processing tool. The noiseless input audio is then fed to the openSMILE tool to extract high-dimensional features. These features are functions of low-level descriptor (LLD) contours. Specifically, we use the *IS13-ComParE* openSMILE configuration to extract features which are of dimension 6373 for every input audio, a.

After these features are extracted, a simple fully-connected neural network is trained to reduce the dimension to 300. Finally, we obtain a feature vector, a_f, of dimension 300 for an input audio, a.

Micro-Expression Features Veronica et al. manually annotated facial expressions and use binary features derived from the ground truth annotations to predict deceptive behavior. Facial micro-expressions are also considered to play an important role in detecting deceptive behavior. The data provided by [14] contains 39 facial micro-expressions such as frowning, smiling, eyebrows raising, etc. These are binary features and taken as a feature vector, m_p of dimension 39.

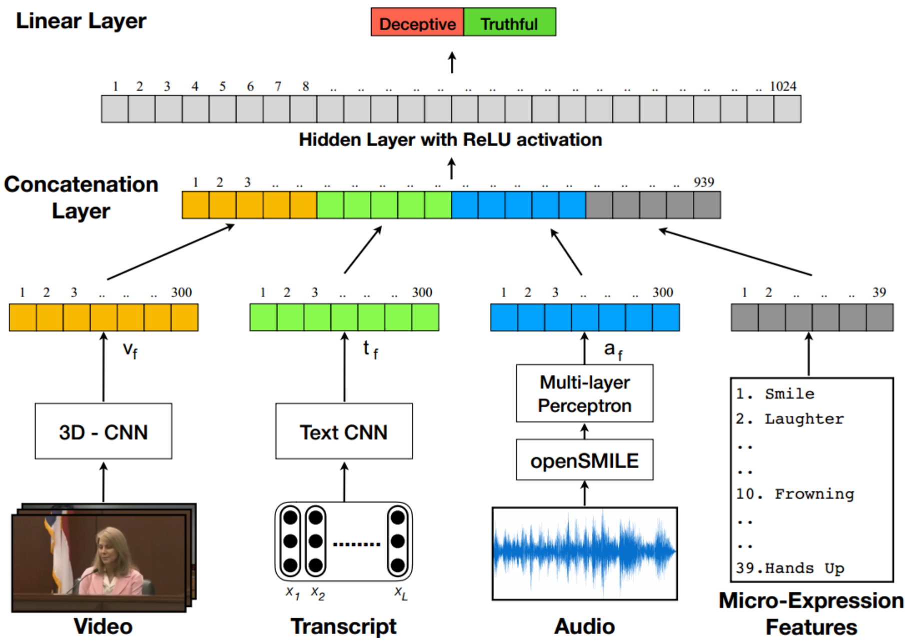

Fig. 1. Architecture of model MLP_C

3.2 Model Description

Multi-modal Model We use a simple Multi-Layer perceptron (MLP) (cite) with hidden layer of size 1024 followed by a linear output layer. We use the rectified linear unit (ReLU) activation function [21] for non-linearity at the hidden layer. A dropout [24] of keep probability, $p = 0.5$, is applied to the hidden layer for regularization.

Figure 1 and 2 shows the architecture of our models, MLP_C and MLP_{H+C}.

Unimodal Models We perform the evaluation on individual modalities and use the same architecture as multi-modal model. The only difference is that the input is either v_f, or a_f, or t_f, or m_f. Hence no data fusion is performed. We name the model as MLP_U.

3.3 Data Fusion

We fuse the features from individual modalities to map them into a joint space. To achieve this, we try different kinds of data fusion techniques:

Concatenation In this method, the features from all the modalities are simply concatenated into a single feature vector. Thus, the extracted features, t_f, a_f, v_f

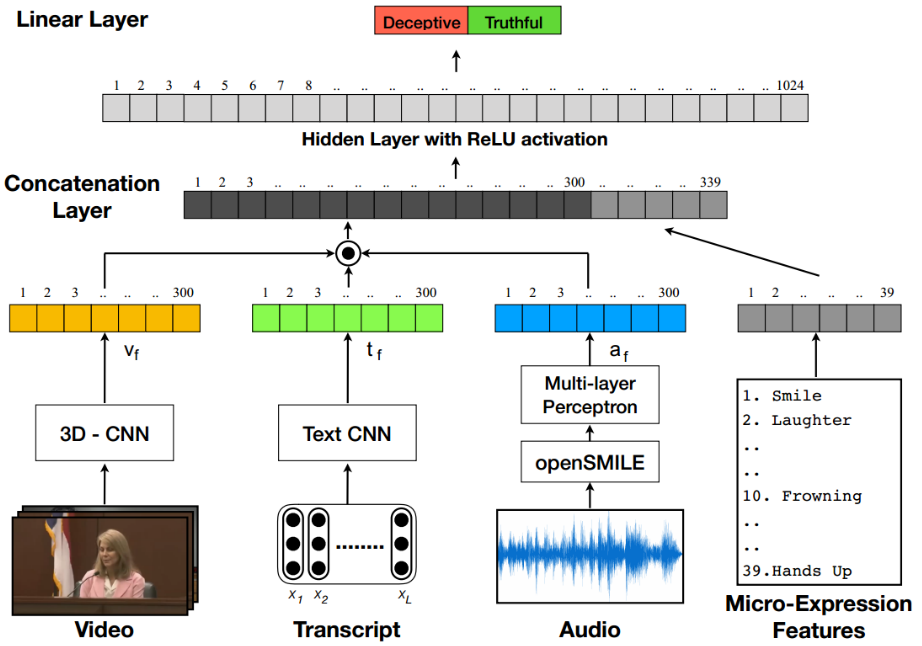

Fig. 2. Architecture of model MLP_{H+C}

and m_f, are simply concatenated to form the representation: $z_f = [t_f; a_f; v_f; m_f]$ of dimension $d_{in} = 939$.

We call this model configuration as MLP_C and is shown in the figure 1.

Hadamard + Concatenation In this method, the audio features, visual features and, textual features are fusion by using hadamard product. Then the Micro-Expression features are concatenated with the product. Thus, we have $z_f = [t_f \odot a_f \odot v_f; m_f]$ of dimension $d_{in} = 339$, where $(A \odot B)$ is element-wise multiplication between matrices A and B (also known as Hadamard product).

We call this model configurtion as MLP_{H+C} and is shown in the figure 2.

3.4 Loss

The training objective is to minimise the cross-entropy between the model's output and the true labels. We trained the models with back-propagation using Stochastic Gradient Descent optimizer. The loss function is:

$$J = \frac{-1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{i,j}\ log_2(\hat{y}_{i,j}) \tag{1}$$

Here, N is the number of samples, C is the number of categories (in our case, $C = 2$). y_i is the one-hot vector ground truth of i^{th} sample and $\hat{y}_{i,j}$ is its predicted probability of belonging to class j.

4 Experiment

4.1 Data

For evaluating our deception detection model, we use a real-life deception detection dataset by [14]. This dataset contains 121 video clips of courtroom trials. Out of these 121 videos, 61 of them are of deceptive nature while the remaining 60 are of truthful nature.

The dataset contains multiple videos from one subject. In order to avoid bleeding of personalities between train and test set, we perform a 10-fold cross validation with subjects instead of videos as suggested by Wu *et al.* [15]. This ensures that videos of the same subjects are not in both training and test set.

4.2 Baselines

Wu *et al.* [15] have made use of various classifiers such as Logistic Regression (LR), Linear SVM (L-SVM), Kernel SVM (K-SVM). They report the AUC-ROC values obtained by the classifiers for different combination of modalities. We compare the AUC obtained by our models against only Linear SVM (L-SVM) and Logistic Regression (LR).

Pérez-Rosas *et al.* [14] use Decision Trees (DT) and Random Forest (RF) as their classifiers. We compare the accuracies of our models against DT and RF.

Table 1. Comparision of AUC

Features	L-SVM [15]	LR [15]	MLP_U	MLP_C	MLP_{H+C}
Random	-	-	0.4577	0.4788	0.4989
Audio	**0.7694**	0.6683	0.5231	-	-
Visual	0.7731	0.6425	**0.9596**	-	-
Textual (Static)	0.6457	0.5643	**0.8231**	-	-
Textual (Non-static)	-	-	**0.9455**	-	-
Micro-Expression	0.7964	**0.8275**	0.7512	-	-
All Features (Static)	0.9065	0.9221	-	0.9033	**0.9348**
All Features (Non-static)	-	-	-	0.9538	**0.9799**

5 Results

Tables 1 & 2 presents the performances of MLP and its variants along with the state-of-the-art models. During feature extraction from text, we train the TextCNN model with two different settings: one, by keeping the word vector representation static; two, by optimizing the vector along with the training (non-static). In our results, we also show the performance of the model with these two textual features separately. Additionally, we also mention the results we got from our models for feature vectors initialized with random numbers.

Table 1 shows that our model, MLP_{H+C}, obtains an AUC of 0.9799 while outperforming all other competitive baselines with a huge margin.

Table 2. Comparing accuracies of our model with baselines

Features	DT [14]	RF [14]	MLP_U	MLP_C	MLP_{H+C}
Random	-	-	43.90%	45.32%	48.51%
Audio	-	-	**52.38%**	-	-
Visual	-	-	**93.08%**	-	-
Textual (Static)	60.33%	50.41%	**80.16%**	-	-
Textual (Non-static)	-	-	**90.24%**	-	-
Micro-Expression	68.59%	73.55%	**76.19%**	-	-
All Features (Static)	75.20%	50.41%	-	90.49%	**90.99%**
All Features (Non-static)	-	-	-	95.24%	**96.14%**

Table 2 compares the performance our models with Decision Tree and Linear Regression models [14]. Our model, MLP_{H+C}, again outperforms other baselines by achieving an accuracy of 96.14%. We can also infer that visual and textual features play a major role in the performance of our models; followed by Micro-Expressions and audio. This conforms with the findings by [14] that facial display features and unigrams contribute the most to detecting deception.

As we can see that, our approach outperforms the baselines by a huge margin. Our neural models simple and straightforward, hence the results show that right feature extraction methods can help in unveiling significant signals that are useful for detecting deceptive nature.

6 Drawbacks

Though our models outperform the previous state-of-the-art models, we still acknowledge the drawbacks of our approach as follows:

- Our models still rely on a small dataset with only 121 videos. Due to this, our models are prone to over-fitting if not carefully trained with proper regularization.

– Also, due to the limited scenarios in the dataset, the model may not show the same performance for out-of-domain scenarios.

7 Conclusions and Future Work

In this paper, we presented a system to detect deceptive nature from videos. Surprisingly, our model performed well even with only 121 videos provided in the dataset, which is generally not a feasible number of data points for neural models. As a result, we conclude that there exists a certain pattern in the videos that provide highly important signals for such precise classification. We performed various other evaluations not presented in this paper, to confirm the performance of our model. From these experiments, we observed that visual and textual features predominantly contributed to accurate predictions followed by Micro-Expression features. Empirically, we observed that our model MLP_{H+C} converged faster in comparison with MLP_C.

While our system performs well on the dataset by [14], we can not claim the same performance of our model for larger datasets covering a larger number of environments into consideration. Hence, creating a large multi-modal dataset with a large number of subjects under various environmental condition is part of our future work. This would pave a way to build more robust and efficient learning systems for deception detection. Another interesting path to explore is detecting deception under social dyadic conversational setting.

References

1. Bond Jr, C.F., DePaulo, B.M.: Accuracy of deception judgments. Personality and Social Psychology Review **10** (2006) 214–234
2. DePaulo, B.M., Lindsay, J.J., Malone, B.E., Muhlenbruck, L., Charlton, K., Cooper, H.: Cues to deception. Psychological bulletin **129** (2003) 74
3. Smith, P.K., Mahdavi, J., Carvalho, M., Fisher, S., Russell, S., Tippett, N.: Cyberbullying: its nature and impact in secondary school pupils. Journal of Child Psychology and Psychiatry **49** (2008) 376–385
4. Hinduja, S., Patchin, J.W.: Bullying beyond the schoolyard: Preventing and responding to cyberbullying. Corwin Press (2014)
5. Baptiste Su, J.: France to impose restrictions on facebook, twitter in fight against fake news during elections (2018) [Online; posted 09-January-2018].
6. Mihalcea, R., Pulman, S.G.: Linguistic ethnography: Identifying dominant word classes in text. In: CICLing, Springer (2009) 594–602
7. Pennebaker, J.W., Francis, M.E., Booth, R.J.: Linguistic inquiry and word count: Liwc 2001. Mahway: Lawrence Erlbaum Associates **71** (2001) 2001
8. Yancheva, M., Rudzicz, F.: Automatic detection of deception in child-produced speech using syntactic complexity features. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics (2013) 944–953
9. Vrij, A.: Detecting lies and deceit: The psychology of lying and implications for professional practice. Wiley (2000)

10. Gannon, T.A., Beech, A.R., Ward, T.: Risk assessment and the polygraph. The use of the polygraph in assessing, treating and supervising sex offenders: A practitioner's guide (2009) 129–154
11. Ekman, P.: Telling lies: Clues to deceit in the marketplace, politics, and marriage (revised edition). WW Norton & Company (2009)
12. Caso, L., Maricchiolo, F., Bonaiuto, M., Vrij, A., Mann, S.: The impact of deception and suspicion on different hand movements. Journal of Nonverbal behavior **30** (2006) 1–19
13. Cohen, D., Beattie, G., Shovelton, H.: Nonverbal indicators of deception: How iconic gestures reveal thoughts that cannot be suppressed. Semiotica **2010** (2010) 133–174
14. Pérez-Rosas, V., Abouelenien, M., Mihalcea, R., Burzo, M.: Deception detection using real-life trial data. In: Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, ACM (2015) 59–66
15. Wu, Z., Singh, B., Davis, L.S., Subrahmanian, V.: Deception detection in videos. arXiv preprint arXiv:1712.04415 (2017)
16. Ji, S., Xu, W., Yang, M., Yu, K.: 3d convolutional neural networks for human action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **35** (2013) 221–231
17. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 4489–4497
18. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)
19. Kalchbrenner, N., Grefenstette, E., Blunsom, P.: A convolutional neural network for modelling sentences. arXiv preprint arXiv:1404.2188 (2014)
20. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems. (2013) 3111–3119
21. Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10). (2010) 807–814
22. Eyben, F., Weninger, F., Gross, F., Schuller, B.: Recent developments in opensmile, the munich open-source multimedia feature extractor. In: Proceedings of the 21st ACM International Conference on Multimedia. MM '13, New York, NY, USA, ACM (2013) 835–838
23. Norskog, L.: Sound exchange. http://sox.sourceforge.net/ (1991)
24. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research **15** (2014) 1929–1958