Journal for Educators, Teachers and Trainers
The PROFESIOLab electronic, peer-reviewed, open-access Magazine

ISSN 1989-9572

DOI:10.47750/jett.2023.14.05.056

AI DRIVEN DETECTION OF DECEPTION IN ONLINE INTERACTIONS: STRENGTHENING CYBERSECURITY THROUGH BEHAVIOURAL ANALYSIS

B.Hema[1],
Medasani Nagaraju[1],
Sai Kumar Rapolu[1]

Journal for Educators, Teachers and Trainers, Vol.14(5)

https://jett.labosfor.com/

Date of reception: 18 July 2023

Date of revision: 09 Aug 2023

Date of acceptance: 21 Sep 2023

B.**Hema, Medasani Nagaraju, Sai Kumar Rapolu (2023). AI DRIVEN DETECTION OF DECEPTION IN ONLINE INTERACTIONS: STRENGTHENING CYBERSECURITY THROUGH BEHAVIOURAL ANALYSIS. Journal for Educators, Teachers and Trainers, Vol. 14(5). 626-636.**

Journal for Educators, Teachers and Trainers
The PROFESIOLab electronic, peer-reviewed, open-access Magazine

Journal for Educators, Teachers and Trainers, Vol. 14(5)

ISSN 1989 –9572

https://jett.labosfor.com/

AI DRIVEN DETECTION OF DECEPTION IN ONLINE INTERACTIONS: STRENGTHENING CYBERSECURITY THROUGH BEHAVIOURAL ANALYSIS

B.Hema[1], Medasani Nagaraju[1], Sai Kumar Rapolu[1]
[1]Department of Computer Science Engineering,
[1]Sree Dattha Group of Institutions, Sheriguda, Hyderabad, Telangana

ABSTRACT
The widespread usage of the internet has made online interactions an essential part of modern communication. However, the rise in deceptive practices like identity theft, fraud, and misinformation has also coincided with the expansion of digital interactions. To maintain integrity and confidence in online communities, it is increasingly essential to recognize and deal with these dishonest tactics. The primary challenge is developing a dependable, automated system that can identify false information among the thousands of online conversations. In the lack of advanced AI-based solutions, deception detection in online interactions has mostly relied on human monitoring, rule-based algorithms, and keyword-based filters. These conventional methods' limited effectiveness stems from their incapacity to adapt to the development of deceptive techniques and their tendency to provide false positives or negatives. As a result, the demand for effective deception detection systems in online interactions has never been higher. Since social media, e-commerce, and other online forums have grown in popularity, there is now a context in which acting dishonestly can have far-reaching consequences. These platforms need to be dependable and secure for user confidence, cybersecurity, and the overall wellbeing of online communities. Therefore, the purpose of this research is to develop a powerful tool that uses behavioral pattern recognition, advanced linguistic analysis, and machine learning algorithms to consistently discriminate between genuine and fraudulent online encounters. The proposed approach integrates feature engineering and multi-modal techniques to enhance the precision and effectiveness of deception detection in digital communities. In the end, this would offer a more dependable and secure online environment.

Keywords: Communication, Cyberbullying, FeatureEngineering, Artificial Intelligence, Cybersecurity, Machine Learning.

1. INTRODUCTION

The exploration of detecting deception in online interactions stems from the rapid evolution and widespread integration of the internet into modern communication. As online interactions became ubiquitous, so did the emergence of deceptive practices, posing significant threats ranging from misinformation and fraud to identity theft and cyberbullying. The escalating prevalence of these dishonest behaviors has elevated the urgency to develop effective methods for identifying and mitigating them to maintain trust and integrity in digital communities.

Historically, the challenge of deception detection in online interactions was primarily addressed through manual monitoring, keyword-based filters, and rule-based algorithms. However, these conventional methods demonstrated limitations in their adaptability to evolving deceptive tactics, often resulting in either false positives or false negatives. The absence of advanced AI-based systems meant that the effectiveness of online deception detection was hampered, leaving digital platforms vulnerable to deceptive practices.

The rise of social media, e-commerce platforms, and various online forums further exacerbated the challenges, as deceptive practices carried the potential for far-reaching consequences in terms of user confidence, cybersecurity, and the overall well-being of online communities. Recognizing the pressing need for more robust and automated deception detection systems, this research has emerged to address the deficiencies of existing methods.

2. LITERATURE SURVEY

There has been a long history of human interest in identifying deceptive behaviour. Trovillo (1939) addressed the historic evidence date back to the Hindu Dharmasastra of Gautama (900 – 600 BCE) and the Greek philosopher Diogenes (412 – 323 BCE). In 1921, Larson invented the Polygraph (Larson et al., 1932), which has been considered as one of the popular methods for lie detection and works by measuring physiological changes in a person in accordance with stress factors. Typically, the polygraph instrument captures physiological changes such as pulse rate, blood pressure and respiration that can be interpreted by psychological experts to identify truthful or deceptive behaviour. With respect to different scenarios, a polygraph test takes up to four hours which leads to limitations on it use in real time conditions. Research studies have been supporting the validity of the polygraph as well as criticizing its use in specific cases. A meta-study by Axe et al., (Axe et al., 1985) found 10 studies from a pool of 250 (that were sufficiently rigorous to be included), indicated that the controlled question test could perform significantly better than chance under specified narrow conditions. However, the deception classification contained a high number of false positives, false negatives and inconclusive instances. In addition, substantial information about the interviewee's background (e.g. occupation, work record and criminal record) was required to be captured before the examination in order to construct a good set of control questions.

Vocal cues, voice stress and acoustic features have also been employed as indicators to distinguish the act of deceit (Hirschberg, 2005). Distinctive additional micro tremors appear due to cognitive overload during the deceptive behaviour (Walczyk, 2013). However, the performance of deception detection using voice stress analysis has been described as "charlatanry" (Eriksson & Lacerda, 2007). Likewise, linguistics has also investigated the changes in language and its structure to classify signs of deception. Linguistic inquiry and word count analysis for deception detection revealed that truth tellers' statements contain more first-person pronouns and self-references (e.g. mine, our) while liars statements contain more words referring to certainty (e.g. totally, truly) and to other- references (they, themselves) (Eriksson & Lacerda, 2007; Abouelenien et al., 2017). A variety of statistical features including mean length of sentence, mean length of clause and clauses per sentence have been extracted from transcribed interviews to evaluate the linguistic hypothesis that liars use less complex and less detailed sentences.

Vrij et al., (Vrij, 2009) reported on the use of thermal imaging of the facial periorbital area to analyse the variations in blood flow specifically when answering unexpected questions. A thermal facial pattern-based approach introduced by (Pavlidis et al., 2002) claims the deception detection accuracy is comparable to that of polygraph tests. Likewise, a thermodynamic model of blood flow variations using the thermal images of facial periorbital area to detect the deceptive behaviour is presented in (Pavlidis and Levine, 2001, Pavlidis et al., 2002). Relationships between different facial emotions (such as stress, fear, and excitement) and deceptive behaviour using thermal imaging is addressed in (Merla and Romani, 2007). Basher and Reyer, 2014) used thermal variation monitoring of the periorbital region and a nearest neighbor classifier that was trained on a high-dimensional feature vector extracted using an average value from each sub-region to detect deception. Experimental results indicated that the classification accuracy did not differ significantly from a random chance distribution based on leave-one-person-out methodology and five-fold cross validation.

In addition to the aforementioned methods, analysis of eye interactions and facial micro-expressions also have been studied as a non-verbal deception detection method (Ekman, 2001). During the act of deceit, relatively short involuntary facial expressions may appear that can be helpful to detect deceptive behaviour. Furthermore, the analysis of facial expressions in terms of asymmetry and smoothness features (Ekman, 2003) indicate their relationship with the deceptive behaviour. Face orientation and intensity of facial expressions is also used to classify the act of deceit (Tian et al., 2005). Likewise, geometric features (Owayjan, et al., 2012) and micro-expressions (Pfister and Pietikäinen, 2012) extracted from the facial data have also been used to classify the deceptive behaviour. Related research in (Pons and Masip, 2018) indicated the usefulness of facial micro-gestures towards the identification of comprehension levels. Buckingham et al., (2014) used artificial neural networks sequentially to identify the micro-gestures and perform the classification respectively. Pérez-Rosas et al., (Rosas et al., 2015) proposed the multi-model deception detection methodology that used a novel dataset acquired from real public court trials. A variety of linguistic and gesture modalities including facial features were combined together to classify the deceptive behaviour. Results reported a classification accuracy between 65 and 75% with varying combinations of modalities. Furthermore, the results indicated that the system outperformed human experts in terms of correct identification of deceptive behaviour. One of the recent machine-based research studies that uses the direction of gaze, eye movements and blink rate to distinguish the truthful and deceptive behaviours is presented in (Borza et al., 2018). The research outcomes indicated the normalised eye blink rate was an important clue of deception detection. Research carried out in (Marchak, 2013, Nunamaker et al., 2016, Levine, 2014, Schuetzler, 2012, Kumar, 2016, Pak and Zhou, 2011, Lim et al., 2013) also indicate the significance of eye interaction and associated corresponding features towards effective deception detection. Eyes blink rate, pupil dilation and gaze are the most common examples of such a feature set. Research studies indicate the relationship between these attributes and cognitive effort variations in deceptive and truthful subjects (Fukuda, 2001). Like other psychological clues for deception detection, additional cognitive efforts performed by deceivers undergo additional cognitive processes compared to truthful individuals that leads to an increased pupil diameter for deceivers (Proudfoot et al., 2015, Dionisio et al., 2001). In a similar study by Marchak (Marchak, 2013), compared to truthful participants, a suppressed eye blinking rate is noticed for participants involved in a mock crime to transport an explosive device to be used for a disturbance.

3. PROPOSED SYSTEM

3.1 Overview

In response to these challenges. The essence of the AI-driven approach involves training these models on meticulously labeled datasets containing examples of different classes. Through this training process, the models can autonomously learn to extract relevant features from internet users dataset, enabling to discern and classify classes or labels with heightened accuracy.

The provided Python script implements a graphical user interface (GUI) application using Tkinter for a surface identification project based on robot-sensed data. Here's a detailed explanation of the steps carried out by the application:

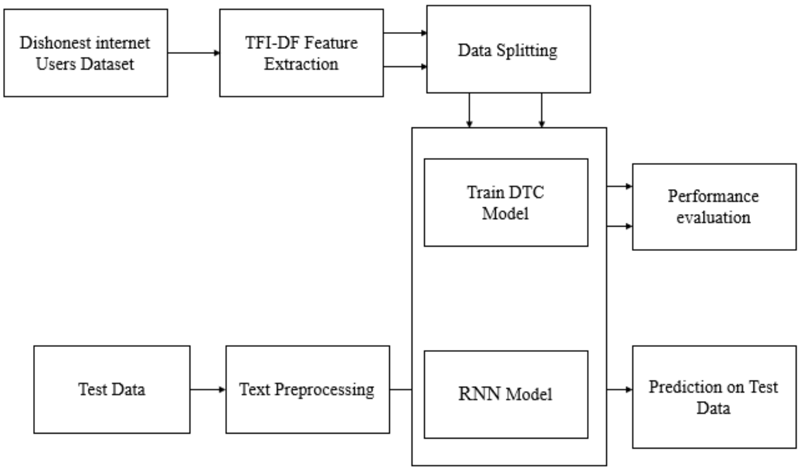
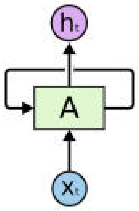


Fig. 1: Block Diagram of Proposed System

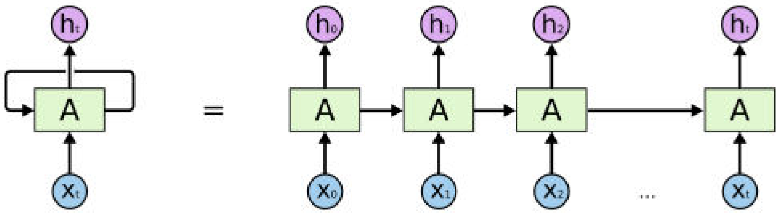
RNN

Recurrent Neural Networks Humans don’t start their thinking from scratch every second. As you read this essay, you understand each word based on your understanding of previous words. You don’t throw everything away and start thinking from scratch again. Your thoughts have persistence. Traditional neural networks can’t do this, and it seems like a major shortcoming. For example, imagine you want to classify what kind of event is happening at every point in a movie. It’s unclear how a traditional neural network could use its reasoning about previous events in the film to inform later ones. Recurrent neural networks address this issue. They are networks with loops in them, allowing information to persist



Recurrent Neural Networks have loops.

In the above diagram, a chunk of neural network, , looks at some input and outputs a value . A loop allows information to be passed from one step of the network to the next. These loops make recurrent neural networks seem kind of mysterious. However, if you think a bit more, it turns out that they aren't all that different than a normal neural network. A recurrent neural network can be thought of as multiple copies of the same network, each passing a message to a successor. Consider what happens if we unroll the loop:

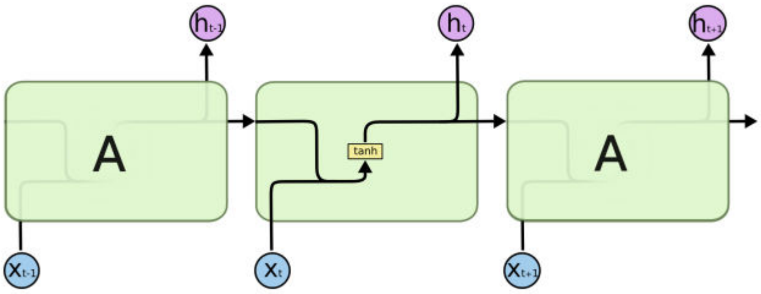


An unrolled recurrent neural network.

This chain-like nature reveals that recurrent neural networks are intimately related to sequences and lists. They're the natural architecture of neural network to use for such data. And they certainly are used! In the last few years, there have been incredible success applying RNNs to a variety of problems: speech recognition, language modeling, translation, image captioning Almost all exciting results based on recurrent neural networks are achieved with them. It's these LSTMs that this essay will explore.

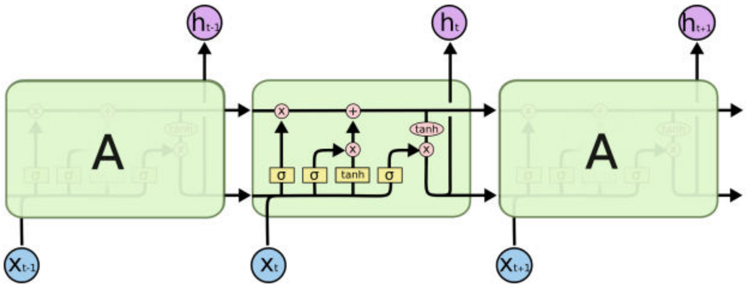
LSTM Networks

Long Short-Term Memory networks – usually just called "LSTMs" – are a special kind of RNN, capable of learning longterm dependencies. They were introduced by Hochreiter &Schmidhuber (1997) (http://www.bioinf.jku.at/publications/older/2604.pdf), and were refined and popularized by many people in following work. They work tremendously well on a large variety of problems, and are now widely used. LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn All recurrent neural networks have the form of a chain of repeating modules of neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer.

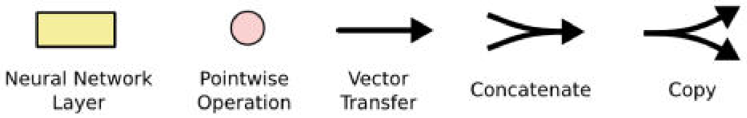


The repeating module in a standard RNN contains a single layer.

LSTMs also have this chain like structure, but the repeating module has a different structure. Instead of having a single neural network layer, there are four, interacting in a very special way.



The repeating module in an LSTM contains four interacting layers.

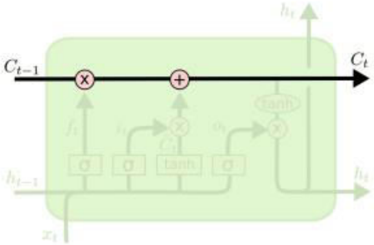


In the above diagram, each line carries an entire vector, from the output of one node to the inputs of others. The pinkcircles represent pointwise operations, like vector addition, while the yellow boxes are learned neural network layers.Lines merging denote concatenation, while a line forking denotes its content being copied and the copies going todifferent locations.

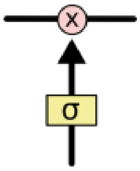
The Core Idea Behind LSTMs

The key to LSTMs is the cell state, the horizontal line running through the top of the diagram.

The cell state is kind of like a conveyor belt. It runs straight down the entire chain, with only some minor linear interactions. It's very easy for information to just flow along it unchanged.



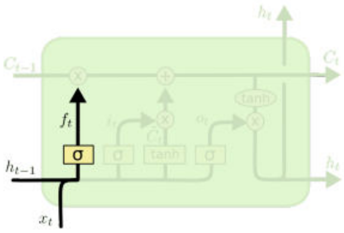
The LSTM does have the ability to remove or add information to the cell state, carefully regulated by structures called gates. Gates are a way to optionally let information through. They are composed out of a sigmoid neural net layer and a pointwise multiplication operation.



The sigmoid layer outputs numbers between zero and one, describing how much of each component should be let through. A value of zero means "let nothing through," while a value of one means "let everything through!" An LSTM has three of these gates, to protect and control the cell state.

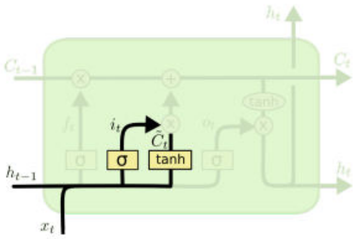
Step-By-Step LSTM Walk Through

The first step in our LSTM is to decide what information we're going to throw away from the cell state. This decision is made by a sigmoid layer called the "forget gate layer." It looks at and , and outputs a number between and for each number in the cell state . A represents "completely keep this" while a represents "completely get rid of this." Let's go back to our example of a language model trying to predict the next word based on all the previous ones. In such a problem, the cell state might include the gender of the present subject, so that the correct pronouns can be used. When we see a new subject, we want to forget the gender of the old subject.



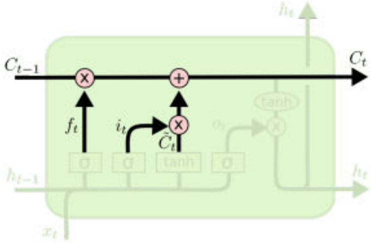
$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

The next step is to decide what new information we're going to store in the cell state. This has two parts. First, a sigmoid layer called the "input gate layer" decides which values we'll update. Next, a tanh layer creates a vector of new candidate values, , that could be added to the state. In the next step, we'll combine these two to create an update to the state.In the example of our language model, we'd want to add the gender of the new subject to the cell state, to replace the old one we're forgetting.



$$i_t = \sigma\left(W_i\cdot[h_{t-1}, x_t] \;+\; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C\cdot[h_{t-1}, x_t] \;+\; b_C)$$

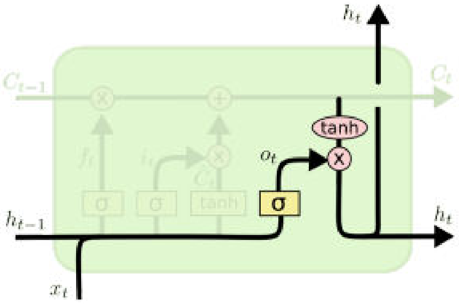
It’s now time to update the old cell state, , into the new cell state . The previous steps already decided what todo, we just need to actually do it.We multiply the old state by , forgetting the things we decided to forget earlier. Then we add . This is the newcandidate values, scaled by how much we decided to update each state value.In the case of the language model, this is where we’d actually drop the information about the old subject’s gender andadd the new information, as we decided in the previous steps.



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Finally, we need to decide what we’re going to output. This output will be based on our cell state, but will be a filteredversion. First, we run a sigmoid layer which decides what parts of the cell state we’re going to output. Then, we put thecell state through (to push the values to be between and) and multiply it by the output of the sigmoid gate,so that we only output the parts we decided to.

For the language model example, since it just saw a subject, it might want to output information relevant to a verb, incase that’s what is coming next. For example, it might output whether the subject is singular or plural, so that we knowwhat form a verb should be conjugated into if that’s what follows next.Variants on Long Short-Term MemoryWhat I’ve described so far is a pretty normal LSTM. But not all LSTMs are the same as the above. In fact, it seemslike almost every paper involving LSTMs uses a slightly different version. The differences are minor, but it’s worthmentioning some of them.



$$o_t = \sigma\left(W_o\;[h_{t-1}, x_t] \;+\; b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

4. RESULTS AND DISCUSSION

Figure 2Graphical User Interface of Dishonest Internet Users with Functionalities Figure represents the graphical user interface designed for detecting dishonest internet users. It likely includes various functionalities related to the identification and analysis of deceptive online behaviour.

Figure 3Preprocessing of the Uploaded Text Dataset showcases the preprocessing steps applied to the uploaded text dataset. This may involve tasks such as cleaning, tokenization, and other text processing techniques.

Figure 4TF-IDF Feature Extraction on Pre-processed Dataset illustrates the application of TF-IDF (Term Frequency-Inverse Document Frequency) feature extraction on the preprocessed dataset. TF-IDF is a technique commonly used in natural language processing for representing text data.

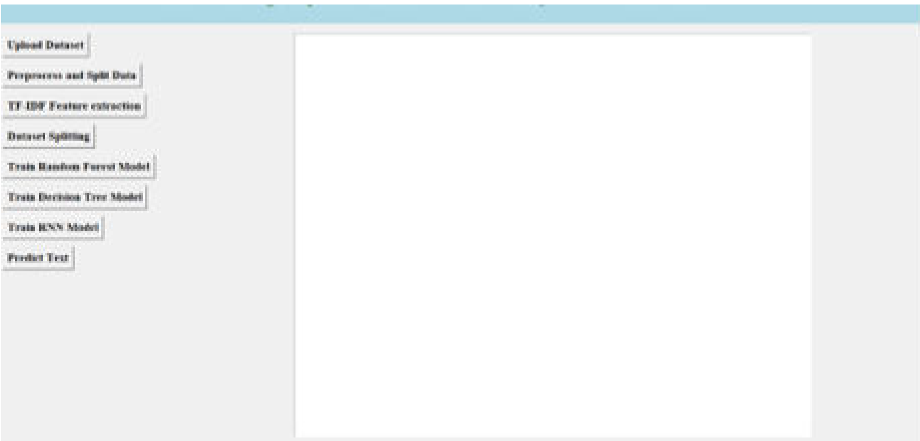
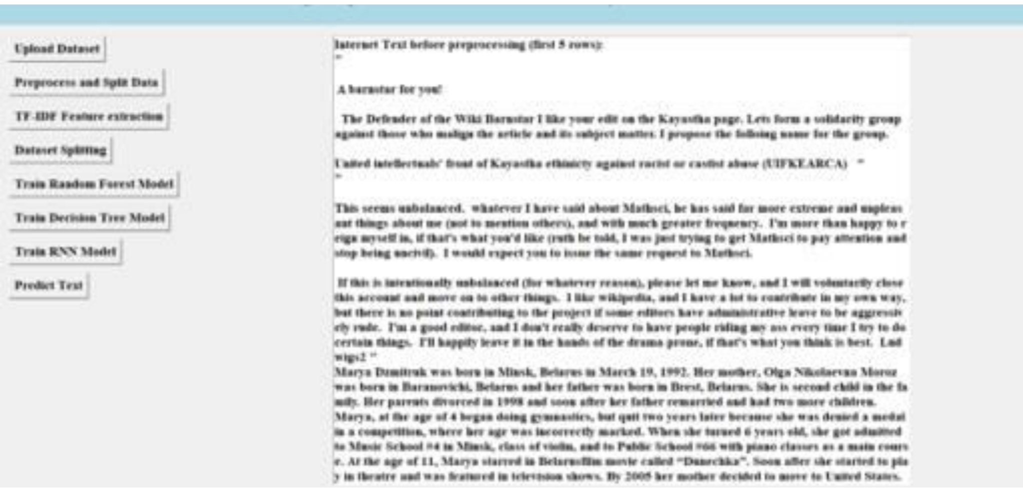
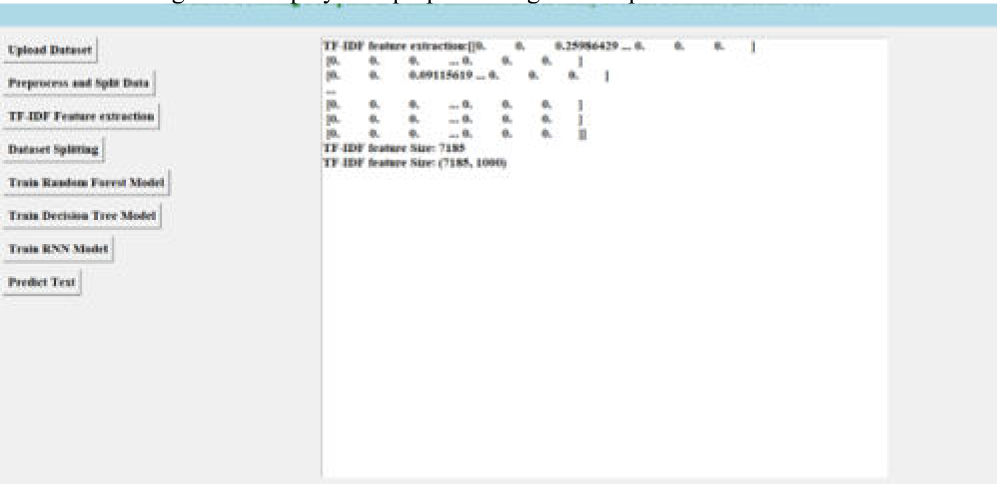

Upload Dataset
Preprocess and Split Data
TF-IDF Feature extraction
Dataset Splitting
Train Random Forest Model
Train Decision Tree Model
Train RNN Model
Predict Text

Figure 2: Represents the graphical user interface of dishonest internet users and it has functionalities.


Upload Dataset
Preprocess and Split Data
TF-IDF Feature extraction
Dataset Splitting
Train Random Forest Model
Train Decision Tree Model
Train RNN Model
Predict Text

```
Internet Text before preprocessing (first 5 rows):
"

A barnstar for you!

 The Defender of the Wiki Barnstar I like your edit on the Kayastha page. Lets form a solidarity group
against those who malign the article and its subject matter. I propose the folloing name for the group.

United intellectuals' front of Kayastha ethinicty against racist or castist abuse (UIFKEARCA)  "
"

This seems unbalanced.  whatever I have said about Mathsci, he has said far more extreme and unpleas
ant things about me (not to mention others), and with much greater frequency.  I'm more than happy to r
eign myself in, if that's what you'd like (truth be told, I was just trying to get Mathsci to pay attention and
stop being uncivil).  I would expect you to issue the same request to Mathsci.

 If this is intentionally unbalanced (for whatever reason), please let me know, and I will voluntarily close
this account and move on to other things.  I like wikipedia, and I have a lot to contribute in my own way,
but there is no point contributing to the project if some editors have administrative leave to be aggressiv
ely rude.  I'm a good editor, and I don't really deserve to have people riding my ass every time I try to do
certain things.  I'll happily leave it in the hands of the drama prone, if that's what you think is best.  Lad
wigs2 "
Marya Dzmitruk was born in Minsk, Belarus in March 19, 1992. Her mother, Olga Nikolaevna Moroz
was born in Baranovichi, Belarus and her father was born in Brest, Belarus. She is second child in the fa
mily. Her parents divorced in 1998 and soon after her father remarried and had two more children.
Marya, at the age of 4 began doing gymnastics, but quit two years later because she was denied a medal
in a competition, where her age was incorrectly marked. When she turned 6 years old, she got admitted
to Music School #4 in Minsk, class of violin, and to Public School #66 with piano classes as a main cours
e. At the age of 11, Marya starred in Belarusfilm movie called "Danechka". Soon after she started to pla
y in theatre and was featured in television shows. By 2005 her mother decided to move to United States.
```

Figure 3: Displays the preprocessing of the uploaded text dataset.


Upload Dataset
Preprocess and Split Data
TF-IDF Feature extraction
Dataset Splitting
Train Random Forest Model
Train Decision Tree Model
Train RNN Model
Predict Text

```
TF-IDF feature extraction:[[0.        0.        0.25986429 ... 0.        0.        0.        ]
 [0.        0.        0.        ... 0.        0.        0.        ]
 [0.        0.        0.09115619 ... 0.        0.        0.        ]
 ...
 [0.        0.        0.        ... 0.        0.        0.        ]
 [0.        0.        0.        ... 0.        0.        0.        ]
 [0.        0.        0.        ... 0.        0.        0.        ]]
TF-IDF feature Size: 7185
TF-IDF feature Size: (7185, 1000)
```

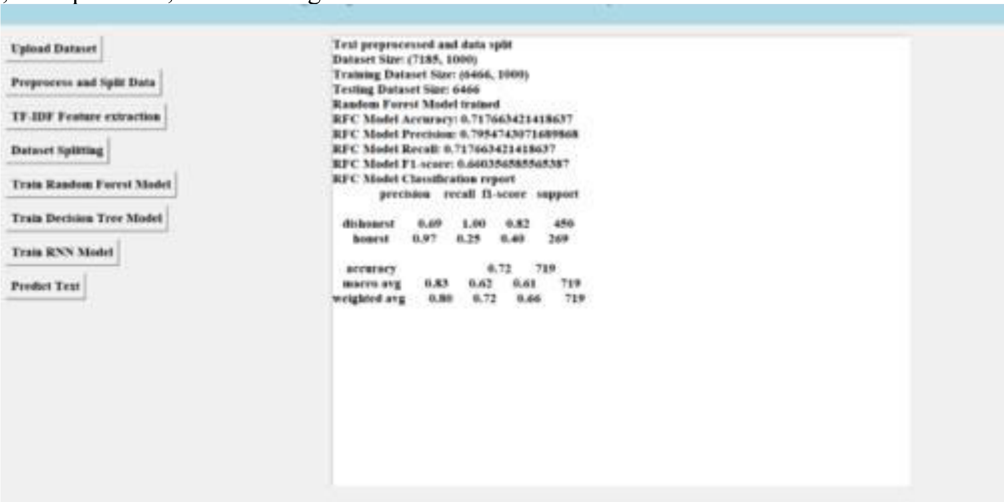
Figure 4: Shows the application of TF-IDF feature extraction on preprocessed dataset.

Figure 5 Performance Metrics of Random Forest Classifier Figure 4 displays the application of performance metrics for a Random Forest Classifier. This may include metrics such as accuracy, precision, recall, and F1-score, providing an assessment of the classifier's effectiveness.

Figure 6 Performance Metrics of Decision Tree Classifier Similar to Figure 4, Figure 5 presents the application of performance metrics, but specifically for a Decision Tree Classifier. It offers insights into the performance of the Decision Tree model.

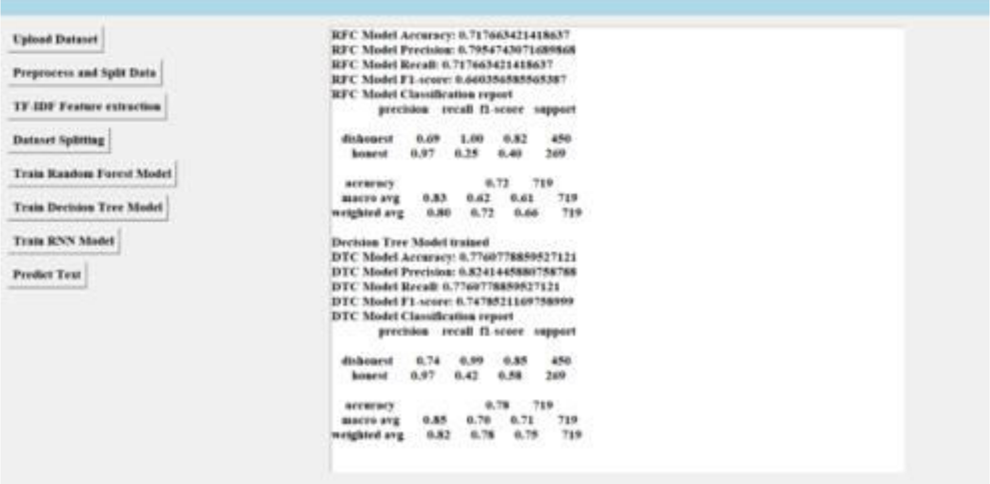
Figure 7 Performance Metrics of Recurrent Neural Network (RNN) showcases the application of performance metrics for a Recurrent Neural Network (RNN). This type of neural network is often used for sequence-based data and may have different evaluation criteria compared to traditional classifiers.Figure 8 Confusion Matrix of All Three Models displays the confusion matrix for all three models (Random Forest, Decision Tree, and RNN).

The confusion matrix provides a detailed breakdown of model predictions, including true positives, true negatives, false positives, and false negatives.


Upload Dataset
Preprocess and Split Data
TF-IDF Feature extraction
Dataset Splitting
Train Random Forest Model
Train Decision Tree Model
Train RNN Model
Predict Text

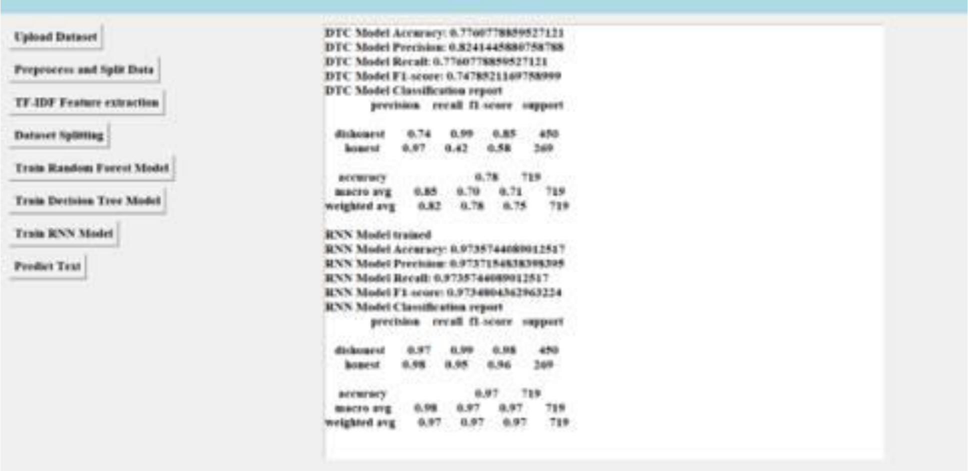
```
Text preprocessed and data split
Dataset Size: (7185, 1000)
Training Dataset Size: (6466, 1000)
Testing Dataset Size: 6466
Random Forest Model trained
RFC Model Accuracy: 0.717663421418637
RFC Model Precision: 0.7954743071689868
RFC Model Recall: 0.717663421418637
RFC Model F1-score: 0.6603565855565387
RFC Model Classification report
          precision    recall  f1-score   support

 dishonest      0.69      1.00      0.82       450
    honest      0.97      0.25      0.40       269

  accuracy                          0.72       719
 macro avg      0.83      0.62      0.61       719
weighted avg    0.80      0.72      0.66       719
```

Figure 5: shows the application of performance metrics of Random Forest Classifier


Upload Dataset
Preprocess and Split Data
TF-IDF Feature extraction
Dataset Splitting
Train Random Forest Model
Train Decision Tree Model
Train RNN Model
Predict Text

```
RFC Model Accuracy: 0.717663421418637
RFC Model Precision: 0.7954743071689868
RFC Model Recall: 0.717663421418637
RFC Model F1-score: 0.6603565855565387
RFC Model Classification report
          precision    recall  f1-score   support

 dishonest      0.69      1.00      0.82       450
    honest      0.97      0.25      0.40       269

  accuracy                          0.72       719
 macro avg      0.83      0.62      0.61       719
weighted avg    0.80      0.72      0.66       719

Decision Tree Model trained
DTC Model Accuracy: 0.7760778859527121
DTC Model Precision: 0.8241445880758788
DTC Model Recall: 0.7760778859527121
DTC Model F1-score: 0.7478521169758999
DTC Model Classification report
          precision    recall  f1-score   support

 dishonest      0.74      0.99      0.85       450
    honest      0.97      0.42      0.58       269

  accuracy                          0.78       719
 macro avg      0.85      0.70      0.71       719
weighted avg    0.82      0.78      0.75       719
```

Figure 6: shows the application of performance metrics of Decision Tree Classifier


Upload Dataset
Preprocess and Split Data
TF-IDF Feature extraction
Dataset Splitting
Train Random Forest Model
Train Decision Tree Model
Train RNN Model
Predict Text

```
DTC Model Accuracy: 0.7760778859527121
DTC Model Precision: 0.8241445880758788
DTC Model Recall: 0.7760778859527121
DTC Model F1-score: 0.7478521169758999
DTC Model Classification report
          precision    recall  f1-score   support

 dishonest      0.74      0.99      0.85       450
    honest      0.97      0.42      0.58       269

  accuracy                          0.78       719
 macro avg      0.85      0.70      0.71       719
weighted avg    0.82      0.78      0.75       719

RNN Model trained
RNN Model Accuracy: 0.9735744089012517
RNN Model Precision: 0.9737154838398395
RNN Model Recall: 0.9735744089012517
RNN Model F1-score: 0.9734804362963224
RNN Model Classification report
          precision    recall  f1-score   support

 dishonest      0.97      0.99      0.98       450
    honest      0.98      0.95      0.96       269

  accuracy                          0.97       719
 macro avg      0.98      0.97      0.97       719
weighted avg    0.97      0.97      0.97       719
```

Figure 7: shows the application of performance metrics of Recurrent Neural Network

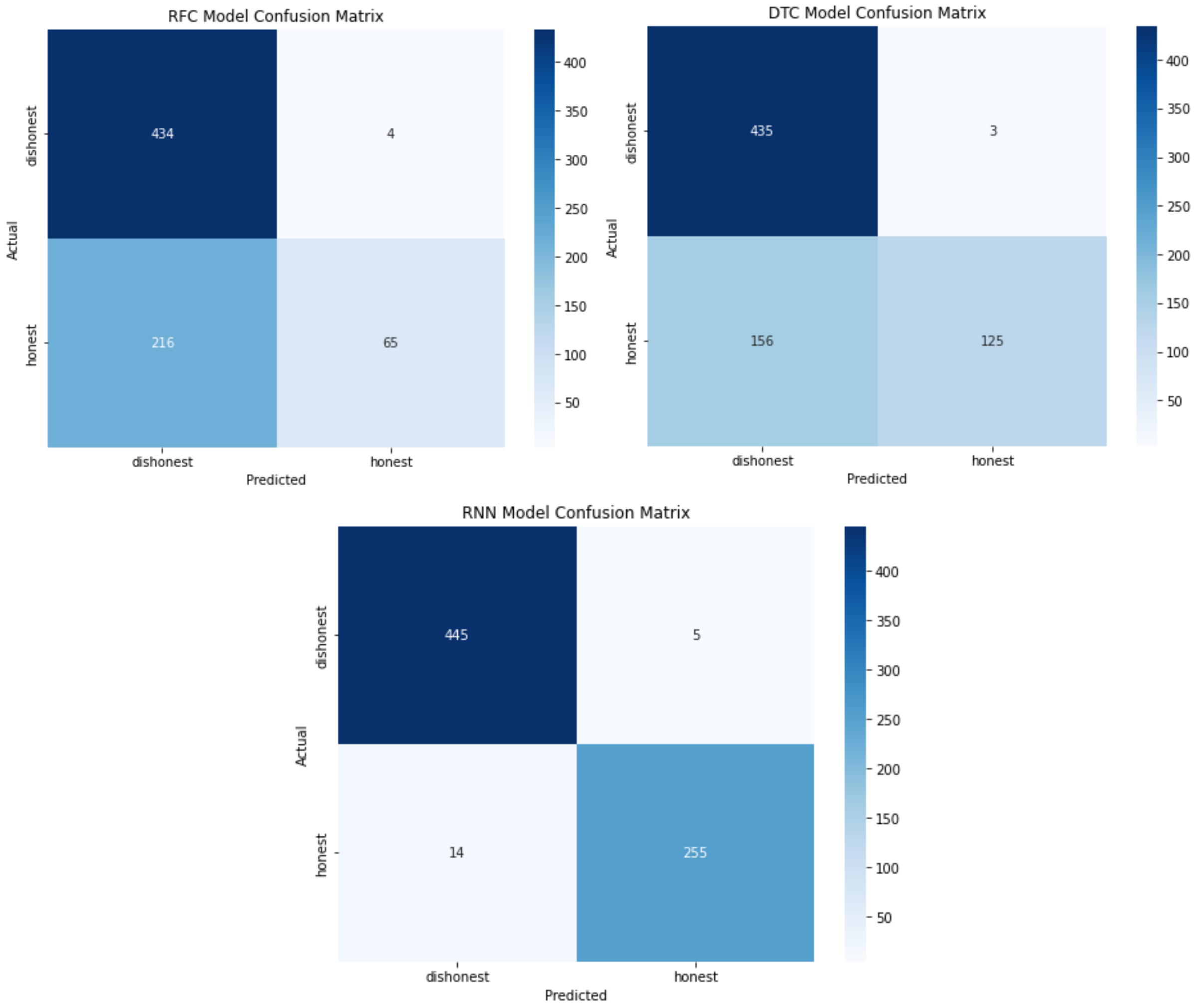
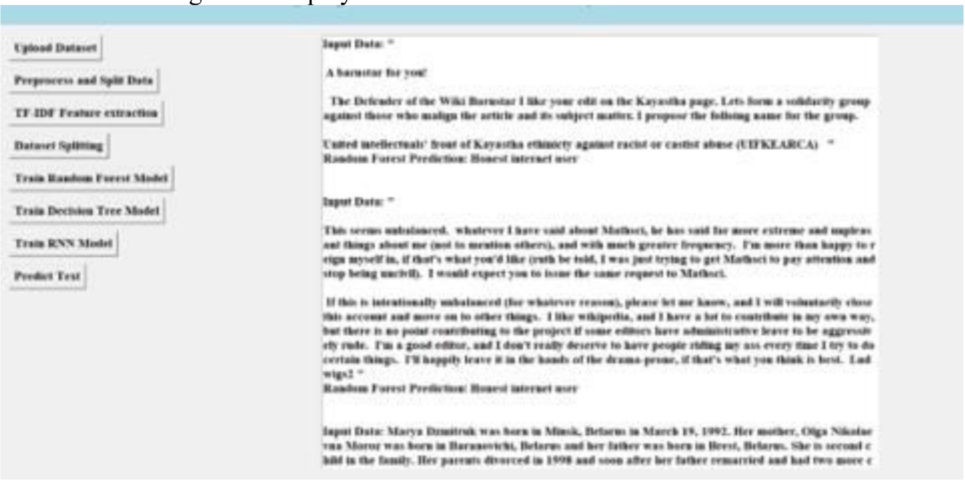


Figure 8: Displays the confusion matrix of All three model.


Upload Dataset
Perprocess and Split Data
TF-IDF Feature extraction
Dataset Splitting
Train Random Forest Model
Train Decision Tree Model
Train RNN Model
Predict Text

Input Data: "

A barnstar for you!

The Defender of the Wiki Barnstar I like your edit on the Kayastha page. Lets form a solidarity group against those who malign the article and its subject matter. I propose the folloing name for the group.

United intellectuals' front of Kayastha ethnicity against racist or castist abuse (UIFKEARCA) "
Random Forest Prediction: Honest internet user

Input Data: "

This seems unbalanced. whatever I have said about Mathsci, he has said far more extreme and unpleas ant things about me (not to mention others), and with much greater frequency. I'm more than happy to r eign myself in, if that's what you'd like (truth be told, I was just trying to get Mathsci to pay attention and stop being uncivil). I would expect you to issue the same request to Mathsci.

If this is intentionally unbalanced (for whatever reason), please let me know, and I will voluntarily close this account and move on to other things. I like wikipedia, and I have a lot to contribute in my own way, but there is no point contributing to the project if some editors have administrative leave to be aggressiv ely rude. I'm a good editor, and I don't really deserve to have people riding my ass every time I try to do certain things. I'll happily leave it in the hands of the drama-prone, if that's what you think is best. Lad wigs! "
Random Forest Prediction: Honest internet user

Input Data: Marya Dzmitruk was born in Minsk, Belarus in March 19, 1992. Her mother, Olga Nikolae vna Moroz was born in Baranovichi, Belarus and her father was born in Brest, Belarus. She is second c hild in the family. Her parents divorced in 1998 and soon after her father remarried and had two more c

Figure 9: Shows the model predicted outcome on the test data.

Figure 9 Model Predicted Outcome on Test Data shows the predicted outcomes of the models on test data. It may include visualizations or summaries illustrating how well the models perform on unseen data.

Table 1: Performance Comparison of Quality Metrics provides a comprehensive comparison of performance metrics obtained using Decision Tree Classifier, Random Forest Classifier, and Recurrent Neural Network (RNN). It likely includes metrics such as accuracy, precision, recall, and F1-score, enabling a side-by-side assessment of the models.

Table 1: Performance comparison of quality metrics obtained using Decision Tree Classifier,Randomforest classifier model and RNN.

Model	Decision Tree Classifier	Random Forest Classifier	RNN
Accuracy (%)	85	72	97
Precision (%)	70	83	98
Recall (%)	70	62	97
F1-score (%)	71	61	97

5. CONCLUSION

The increasing prevalence of deceptive practices in online interactions necessitates advanced and automated systems to effectively detect and mitigate dishonest behaviors. Traditional methods, relying on manual monitoring and rule-based algorithms, fall short in adapting to the dynamic nature of deceptive tactics in the digital realm. This research addresses this critical challenge by proposing a sophisticated AI-based system for detecting deception in online interactions.

The utilization of machine learning algorithms, advanced linguistic analysis, and behavioral pattern recognition represents a significant advancement in the field of deception detection. By integrating multi-modal approaches and feature engineering, the proposed system aims to enhance accuracy and efficiency. This is crucial for maintaining trust, integrity, and user confidence in the digital communities that have become integral parts of our daily lives.

The research not only acknowledges the urgency of the issue but also proposes a solution that aligns with the technological landscape of modern communication. The importance of fostering a safer and more trustworthy online environment cannot be overstated, considering the far-reaching consequences of deceptive practices on social media, e-commerce, and various online forums.

REFERENCES

[1] Abouelenien et al., 2017M. Abouelenien, V. Pérez-Rosas, R. Mihalcea, M. BurzoDetecting deceptive behavior via integration of discriminative features from multiple modalitiesIEEE Transactions on Information Forensics and Security, 12 (05) (2017), pp. 1042-1055, 10.1109/TIFS.2016.2639344

[2] Abouelenien et al., 2014Abouelenien, M., Rosas, V. P., Mihalcea, R., & Burzo, M. (2014). Deception detection using a multimodal approach. 16th International Conference on Multimodal Interaction (ICMI '14). ACM, New York, NY, USA, 58-65, doi: https://doi.org/10.1145/2663204.2663229.

[3] Aristoklis et al., 2005A.D. Anastasiadis, G.D. Magoulas, M.N. VrahatisNew globally convergent training scheme based on the resilient propagation algorithmNeurocomputing, 64 (2005), pp. 253-270, 10.1016/j.neucom.2004.11.016

[4] Axe et al., 1985L. Axe, D. Dougherty, T. CrossThe validity of polygraph testing: Scientific analysis and public controversyAmerican Psychologist, 40 (03) (1985), pp. 355-366, 10.1037/0003-066X.40.3.355

[5] Basher and Reyer, 2014Bashar, A., & Reyer, Z. (2014). Thermal Facial Analysis for Deception Detection. IEEE Transactions on Information Forensics and Security. 09(06), 1015-1023, doi: 10.1109/TIFS.2014.2317309.

[6] Bond and DePaulo, 2006C.F. Bond Jr., B.M. DePauloAccuracy of Deception JudgmentsPers Soc Psychol Rev, 10 (3) (2006), pp. 214-234, 10.1207/s15327957pspr1003_2

[7] Borza et al., 2018D. Borza, R. Itu, R. DanescuIn the Eye of the Deceiver: Analyzing Eye Movements as a Cue to DeceptionJournal of Imaging, MDPI, 4 (10) (2018), pp. 1-20, 10.3390/jimaging4100120

[8] Bradski, 2000G. BradskiOpenCV LibraryRetrieved from https://docs.opencv.org/master/d2/d42/tutorial_face_landmark_detection_in_an_image.html

[9] Breiman, 2001L. BreimanRandom forestsMachine learning, 45 (01) (2001), pp. 5-32, 10.1023/A:1010933404324

[10] Buckingham et al., 2014F. Buckingham, K. Crockett, Z. Bandar, J. O'SheaFATHOM: A Neural Network-based Non-verbal Human Comprehension Detection System for Learning EnvironmentsIEEE SSCI, Florida, 403–409 (2014), 10.1109/CIDM.2014.7008696

[11] Chen et al., 2018J. Chen, Z. Chen, Z. Chi, H. FuFacial expression recognition in video with multiple feature fusionIEEE Transactions on Affective Computing, 9 (1) (2018), pp. 38-50, 10.1109/TAFFC.2016.2593719

[12] Cortes and Vapnik, 1995C. Cortes, V. VapnikSupport-vector networksMachine Learning, 20 (3) (1995), pp. 273-297, 10.1007/BF00994018

[13] Crockett, et al., 2017Crockett, K. A., et al. (2017). Do Europe's borders need multi-faceted biometric protection, Biometric Technology Today. 07, 5-8, ISSN: 0969-4765.