Distinguishing Cultural Backgrounds Through Social Touch Patterns: A Machine Learning Approach

YIXUAN ZHUANG, University of Twente, The Netherlands

Touch is one of the most fundamental sensations in humans and the most basic action for interacting with others. In different cultural backgrounds, touch has different meanings. Additionally, social touch itself has various roles and functions. This study uses touch-sensitive patches to collect data on how people from different cultural backgrounds touch in various attention-seeking scenarios. To understand the details of social touch and distinguish the touches made by touchers from different cultural backgrounds, the collected data will be utilized to train models using machine learning for time series such as Random Forest or Long Short Term Memory, which could be used for potential future applications.

Additional Key Words and Phrases: Social touch, Cultural Backgrounds, Touch Sensitive Patch, Machine Learning

1 INTRODUCTION

Touch is one of the five classic senses of humans(Haans & IJsselsteijn, 2005)[9]. It is the earliest developed sensory system in humans after birth(Maurer & Maurer, 1988; Montagu, 1986, as cited in Cascio et al., 2019b)[4]. As the earliest way humans experience the world, touch plays a crucial role in how we interact with our surroundings.

Based on the research by Schirmer et al.(2022)[18], this paper defines social touch as gentle physical contact between individuals aimed at socializing, conveying information, or expressing emotions. Social touch is one of the most fundamental forms of human interaction, which, through basic physical contact, serves to alleviate stress, build a sense of solidarity, and convey feelings of love and compassion(Saarinen et al., 2021)[17]. This paper focuses on the attention-seeking type of social touch, which is an important application of social touch. The reason for choosing this focus is the interest in understanding how people from different cultural backgrounds engage in attention-seeking social touch and the future application of it. Furthermore, recognizing that people's application of and tolerance for social touch vary across different cultural backgrounds, this research will also focus on distinguishing how individuals from different cultural backgrounds use social touch in various scenarios.

The Touch Sensitive Patch(TSP) will be used to collect sufficient data. To simulate people being touched in attention-seeking scenarios, the TSP will be placed on a simulated arm and ask participants, after confirming their informed consent and filling out a short survey(see section 4.1), to simulate the social touches they might make based on the scenarios provided.

To differentiate the social touches made by people from various cultural backgrounds, the data collected by the TSP will be used to train a usable model through machine learning methods. This model can potentially aid in developing applications such as emotional companion robots, infant education robots, and more. This study is particularly interested in personalized human-robot interaction and cultural sensitivity in technology. By examining how individuals from different cultural backgrounds engage in social touch across various scenarios, we can develop emotionally intelligent robots that offer personalized and culturally appropriate responses. This approach aims to enhance user satisfaction and acceptance of robotic companions. Moreover, recognizing and adapting to cultural differences in social touch can prevent misunderstandings and increase the effectiveness of human-robot interactions in multicultural environments.

2 PROBLEM STATEMENT

Although there has been research done in the field of touch, how social touch occurs among people from different social backgrounds remains unclear. In this research, the specific details of social touch will be analyzed and trained using machine learning algorithms such as RF or LSTM to ultimately produce a model. This will address the gaps in research on social touch occurring across different cultural backgrounds.

2.1 Research Question

The problem statement will lead to the following research question:

How can data on social touches performed by people from different cultural backgrounds under different attention-seeking scenarios, collected through Touch Sensitive Patch, be trained by a machine learning algorithm to identify differences between social touches?

This main research question(RQ) can be answered with the following sub-questions:

(1) How to use TSP to collect natural, unbiased social touch data for training a model that can distinguish social touches from people of different social backgrounds?
(2) What kind of data can be considered effective for labeling social touches from different cultural backgrounds and how to process them?
(3) Which machine learning algorithm performs best in distinguishing social touches from different cultural backgrounds?

3 RELATED WORK

Saarinen et al.(2021)[17] reviewed articles about social touches under different contexts and concluded that the pleasure derived from social touch is related to various situational psychological factors. Although they provided a comprehensive summary of previous research on social touch, they did not review papers related to cultural background because another paper(Gallace & Spence, 2010)[7] they cited conducted a detailed study on the influence of cultural factors on social touch. However, this paper reviewed aspects such as the timely response to social touch, which is also worth referencing.

TScIT 41, July 5, 2024, Enschede, The Netherlands

As mentioned above, Gallace and Spence(2010)[7] concluded that cultural background is one of the factors to be considered when researching social touch. They pointed out that in some cultures, such as France, people frequently touch each other, whereas in other cultures, such as Japan, people hardly touch each other.

Further research done by Gumtau(2005)[8] explored the relationship between tactile semiotics and cultural backgrounds by haptic box experiment, which was designed to investigate whether certain emotional values are associated with tactile experiences and to explore the cultural and social coding system of touch. The experiment involved a box with ten different textures presented randomly. Participants could feel these textures with one hand while filling out a semantic differential scale with the other, rating 12 polarized word pairs from one to seven. Afterward, they completed a questionnaire on touch memories and associations to gain insight into their touch awareness. The results showed a high correlation between participants' choices, suggesting a cultural system of semiotics, especially with consistent associations for natural materials. This research doesn't provide which cultural backgrounds are involved and does not focus on social touch, which is defined in the introduction1 of this paper.

Another research about cultural backgrounds and social touch is Schirmer et al.(2022)[18] who did research to identify the factors that make touches comfortable among Chinese and Germans. However, the research was focused on touches that comfort people, which is only a function that social touch has. The social touch can do more than just comfort people but as a communication tunnel in socialization. Furthermore, research done by Suvilehto et al. (2019)[21] explores social touch among Japanese and British individuals and finds that it depends on the strength of the emotional bond between the parties involved.

As discussed above, research has been done in the field of social touch to discover the effect of different cultural backgrounds. However, how to identify a social touch as coming from a specific cultural background has not yet been studied in detail. This is important When creating emotional companion robots or baby care robots, it is crucial because people want to be treated in ways that they feel comfortable and accustomed to, that is, to be cared for or helped by someone who shares or understands their cultural background. Identifying social touches from different cultural backgrounds can guide robots to make appropriate touches.

To distinguish social touches from different cultural backgrounds, this paper will study several machine learning algorithms and try to find the one with the best performance, in other words, with the highest accuracy. Research conducted by Wingerden et al. (2014)[23] demonstrated that a feedforward neural network using all feature sets has good performance in distinguishing social touches from different cultural backgrounds. They also suggested exploring other machine learning models for better performance, which is also the goal of this study.

4 METHODOLOGY

In this section, the methodology that was used by this research to answer each sub-research question as mentioned in the previous section(see section 2.1) is discussed in detail.

4.1 On answering sub-RQ1

To address RQ1, a Python application was designed for data collection. This application consists of two parts: the user study web application and the TSP data collection application. Before the data collection process, the TSP is attached to a simulated arm, with one for the left hand and one for the right hand(see figure 1). Sleeves are placed over the TSP to simulate a realistic scenario. Two devices will be used: one running the user study web application and the other running the TSP recording application. The former will be shown to the participants, while the latter will only be visible to the researcher present(see figure 2). This setup ensures that the researcher can monitor the data collection process in real-time to ensure it is functioning correctly. The user study application first presents the participants with a survey page containing the following eight questions:

(1) Age
(2) Handedness
(3) Gender
(4) Cultural backgrounds that the participant grew up in
(5) The most approachable family member of the participant
(6) The frequency that the participant's families touche each others(in Likert scale)[14]
(7) The frequency that the participant touches a stranger(in Likert scale)[14]
(8) The general frequency that the participant thinks people in his/her country touch others(in Likert scale)[14].

Fig. 1. The TSP attached to the simulated arm

Fig. 2. The data collection process setup

Participants can fill out the survey on the web application and submit it, which will upload their answers to the database and generate a random sequence number. This sequence number represents the four Scenarios that the current participant will encounter. This study focuses on two attention-seeking Scenarios: warning and comforting. The warning Scenarios are:

(1) A person is about to be hit by a moving car.
(2) A person is blocking the passage in a busy bar.

The comforting Scenarios are:

(1) A person is crying after receiving a phone call.
(2) A person is injured.

Additionally, the study considers two types of relationships between the toucher and the touchee: the most approachable family member and a stranger. These four Scenarios are randomly paired with different relationships between the toucher and the touchee and presented to the participant in random order. This ensures the generality of the collected data and minimizes the impact of irrelevant variables, such as different people's reactions to different scenarios.

Regarding the data collection process, participants will first read the informed consent form. If they agree to the terms, they need to sign their names. After that, they will see a screen displaying the user study web app interface where they will complete the survey. The researcher will then provide relevant instructions, such as the option to refrain from touching, the requirement to touch the forearm of the simulated arm, and the maximum touch duration of 10 seconds (to prevent unnecessary repeated data from prolonged repetitive actions in certain Scenarios). The researcher will also position the corresponding arm according to the participant's handedness. After these preparations, the participant will see the first Scenario and perform the touch they intend to execute.

The output data from TSP will be read and recorded in the form of arrays in a JSON file, which is named after the participant's ID and the corresponding scenario together with the relationship between the touchers and touchees.

4.2 On answering sub-RQ2

Because this study focuses on the differences in social touch from different cultural backgrounds, the participants' cultural backgrounds are necessary information. To collect this, the survey requires participants to provide the cultural backgrounds that they grew up in. The reason that the question is not asking about the country that the participants grew up in is that participants may be raised in a family that has different cultural backgrounds from the country they lived in.

Furthermore, this study is also interested in the general touch frequency in different cultural backgrounds. To gather this information, two questions in the survey asked the participants about the frequency of their family members touching each other and the frequency of the people of the cultural backgrounds in which they grew up in touch with each other.

Considering that in some cultures, people do not frequently touch each other, participants may choose not to engage in touching during the data collection process. In such cases, the system will generate a JSON file with the 'has touched' field set to false, and the data will be empty. These no-touch files will be analyzed separately from actual touch data.

According to research by FY et al. (2017)[6], Support Vector Machine(SVM) performed best in terms of precision and accuracy for classification, followed by Random Forest(RF) algorithms. Other than these, Karim et al. (2018)[11] pointed out that the combination of long short term memory(LSTM) and Fully convolutional neural networks (FCNs) performed excellent in classifying time series sequences.

The above algorithms were carefully selected by the researchers after a thorough investigation, as they have the potential to excellently distinguish social touches from different cultural backgrounds. The principles, advantages, and disadvantages of these algorithms will be introduced in more detail in the following sections. This study will also use these algorithms to attempt to train an ideal model and compare their reliability, thereby selecting the optimal choice.

This research on social touch explored various machine learning algorithms to classify social touch data based on cultural background. The investigation began with the implementation of an LSTM-FCN model, chosen for its capability to handle sequential data and its widespread use in time-series analysis. Following the LSTM-FCN model, other deep learning architectures, including Random Forest and Support Vector Machine, were experimented with.

5 BACKGROUND

This section will not provide a very detailed explanation of the principles of each algorithm, as this is beyond the scope of this study, but it will offer a brief introduction and the reasons for choosing these algorithms.

5.1 LSTM-FCN

The Recurrent Neural Network (RNN) is known as a good tool for time series prediction. Time series classification, as defined by Karim et al. (2019)[10], is "a supervised learning task that classifies a series of data points that are commonly collected in equal intervals and depicted in a sequential order", which is exactly the form of the social touch data that was collected in this research. However, RNN suffers from the problems of gradient explosion and gradient vanishing, so Long short term memory(LSTM) was developed to address these issues(Sherstinsky, 2020)[19], which makes LSTM a suitable architecture for processing time series data.

Furthermore, LSTM also enhances the performance of Fully Convolutional Networks (FCNs) with a nominal increase in model size and minimal preprocessing of the dataset(Karim et al., 2018b)[11]. The LSTM-FCN architecture is shown in figure 3. To understand how the LSTM-FCN works, we should be aware that the LSTM-FCN has two parts. The FCN part extracts features that are useful for the research from the raw time series data through convolutional layers. These features are then sent to the LSTM, which is good at learning long-term dependencies and handling sequential data. By using both convolutional and recurrent layers, LSTM-FCN can effectively process time series data, making it robust for complex classification tasks(Karim et al., 2018b)[11].

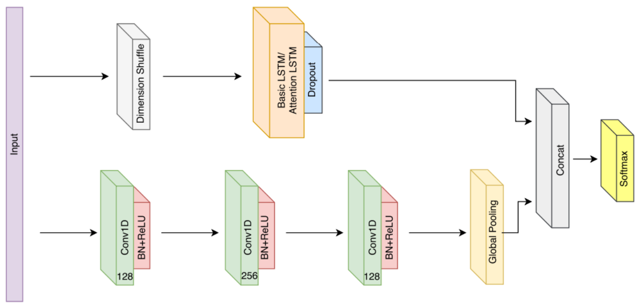


Fig. 3. The LSTM-FCN architecture(Karim et al., 2018b)[11]

5.2 Random Forest

Random Forest is an ensemble classification technique that builds multiple decision trees during training and merges them to improve prediction accuracy. Introduced by Leo Breiman(as cited in Parmar et al., 2018c)[15], it is known for its high accuracy in handling both classification and regression tasks. The method reduces overfitting by using random subsets of data and features, and it can handle missing values efficiently. Despite its higher computational cost and complexity, its ability to provide feature importance makes it valuable for understanding data.

5.3 SVM

Support vector machines (SVM) are machine learning algorithms that learn from empirical data such as samples, measurements, recordings, or observations, also known as kernel machines. The main advantage lies in its solid theoretical foundation and good generalization ability, which can perform well even on high-dimensional and sparse datasets. In addition, SVM is capable of acting as a general approximator for any multivariate function and is particularly suitable for modeling unknown or partially known highly nonlinear, complex systems(Kecman, 2005)[12].

6 RESULT ANALYSE

6.1 Data pre-processing

There are in total 31 participants joined the data collection process which produced 124 data files. Table 1 shows the cultural background distribution among the participants.

As mentioned in section 4.2, if the participants decided not to touch, then the system would record the file with empty data. There are in total 17 out of 124 scenarios in which the participant decided not to perform a touch. Table 2 shows an overview of the result: Participants with a Chinese cultural background are most likely to choose not to touch the touchee, and almost all participants who choose not to touch the touchee do so when the other person is a stranger. However, because the sample size is still too small, this conclusion may be biased.

Because some cultural backgrounds(such as Uzbekistan) have too few samples which might lead to bias in training the model and unbalance in classes, after removing empty data(scenarios that the participant chooses not to touch), the remaining data will be used to extract features and be further classified instead of being directly categorized based on the country. In this study, social touches will be labeled into three classes: low, medium, and high, representing the intensity of the social touch, with high being the highest priority, medium the second, and low the lowest. The intensity is determined based on the average pressure of the touch and its duration. Specifically, for each social touch, the data will be used to calculate the average pressure and the duration, which, along with the relationship between the toucher and the touchee, are considered features. Using these features, the touches are then classified into three categories mentioned above using the k-means clustering algorithm(Wu, 2012)[24], which is an efficient clustering algorithm for machine learning. This study clustered the data points into three clusters, with a random seed set to ensure the reproducibility of the results. Each participant has at least one and at most four social touch data points. Based on the labels of these touches, each participant will be labeled as one of the following classes:

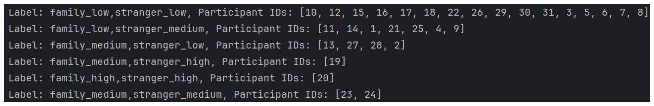
(1) family_low,stranger_low
(2) family_low,stranger_medium
(3) family_low,stranger_high
(4) family_medium,stranger_low
(5) family_medium,stranger_medium
(6) family_medium,stranger_high
(7) family_high,stranger_low
(8) family_high,stranger_medium
(9) family_high,stranger_high

	Total number
Netherlands	11
Moldova	5
China	4
Thailand	2
Africa	1
India	1
International	1
Italy	1
Poland	1
Romania	1
Russia	1
South Korea	1
Uzbekistan	1

Table 1. Overview of participants' cultural backgrounds

	Comforting	Warning	Family	Stranger	in total
China	2	3	2	3	5
Netherlands	3	0	0	3	3
Thailand	2	1	0	3	3
Africa	1	1	0	2	2
India	1	0	1	0	1
Romania	0	1	0	1	1
South Korea	1	0	0	1	1
Taiwan	1	0	0	1	1

Table 2. Not-to-touch in different cultural backgrounds



```
Label: family_low,stranger_low, Participant IDs: [10, 12, 15, 16, 17, 18, 22, 26, 29, 30, 31, 3, 5, 6, 7, 8]
Label: family_low,stranger_medium, Participant IDs: [11, 14, 1, 21, 25, 4, 9]
Label: family_medium,stranger_low, Participant IDs: [13, 27, 28, 2]
Label: family_medium,stranger_high, Participant IDs: [19]
Label: family_high,stranger_high, Participant IDs: [20]
Label: family_medium,stranger_medium, Participant IDs: [23, 24]
```

Fig. 4. Final classification of the participants

The classification is based on the relationship between the toucher and the touchee. If a participant makes a touch labeled as high and another touch labeled as low under a certain relationship, the final label will be medium. In other cases, such as making a touch labeled as high and another labeled as medium, or a touch labeled as low and another labeled as medium, the higher priority label will be assigned. Therefore, theoretically, all participants will be classified into 9 types(as shown above). In practice, due to the limited number of samples, participants were split into the 6 classes shown in figure 4.

After labeling, because the duration of the touch varies, which results in different data lengths, this research uses zero-padding to standardize the length of time series data to solve the problem. After that, the cultural background classes of the touches are labeled using 'LabelEncoder' function in the scikit-learn library[16].

Then, the remaining dataset is split into 80 percent training plus validation sets and 20 percent test sets. The training plus validation set is further split into a 70 percent training set and a 10 percent validation set. The test set will not be used to train the model but only for testing the accuracy of the model. The validation set is used to evaluate the performance of the model during the training process. Another reason to have the validation set is to avoid overfitting. If a model performs well on the training set but is poor on the validation set, it usually indicates overfitting. The data is not randomly assigned to sets. In fact, by using the 'train_test_split' function in scikit-learn library[16] and parameter 'stratify', all subsets maintain the proportion of each class, thereby ensuring all classes are balanced. This is called stratified sampling(Singh & Mangat, 1996)[20]. In this study, in the previous steps, each touch was labeled to indicate its cultural background category. Stratified sampling was performed with this label as the variable, ensuring that the relative frequency of each category remains consistent across each set (i.e., training set, validation set, and test set).

After the above data preprocessing steps, some classes still had relatively few samples. To address this issue, this study applied the Synthetic Minority Over-sampling Technique (SMOTE)(Chawla et al., 2002b)[5], an oversampling technique used to handle class imbalance problems. Specifically, in this study, for classes with fewer samples (e.g., family_high, stranger_high), the nearest neighbor of each sample was found (by default, this technique uses k=5 neighbors, but due to the very small number of samples, k=1 neighbor was chosen), and a new synthetic sample was generated along the line between the original sample and its neighbor. This method generates new samples, making the distribution of minority-class samples more uniform. This study used the SMOTE object from the imbalanced-learn library(Lemaître et al., 2017)[13] to oversample the classes with too few samples. Finally, the data is transformed into a 2D array (samples x features) for the Random Forest (RF) and Support Vector Machine (SVM) models, and into a 3D array (samples x time series length x features) for the LSTM-FCN model. This ensures the data is in the appropriate format for each model type.

6.2 Result of different machine learning algorithms

The model training process uses libraries from TensorFlow[1].

6.2.1 LSTM-FCN. LSTM-FCN model(Karim et al., 2018a)[11] combines Long Short-Term Memory networks (LSTM) and Fully Convolutional Networks (FCN) to handle time series data and perform multi-class classification tasks. This study follows the basic principle of this model and modified it to fit the situation. The model has 2 branches: LSTM branch and FCN branch. The LSTM branch has 3 layers:

(1) Input Layer: The model accepts time series data with a shape of (max_len, num_features).
(2) LSTM Layer: This layer consists of 128 LSTM units, which process the input sequence to capture temporal dependencies.
(3) Dense Layer: Following the LSTM layer, a fully connected layer with 64 units and ReLU activation(Agarap, 2018)[3] is applied to further process the LSTM output.

The FCN branch has a similar structure, an input layer with the same input shape as the LSTM branch, and a convolutional layer that contains 3 layers:

(1) First Layer: 128 filters, kernel size 8, 'same' padding, ReLU activation.
(2) Second Layer: 256 filters, kernel size 5, 'same' padding, ReLU activation.
(3) Third Layer: 128 filters, kernel size 3, 'same' padding, ReLU activation.

Last but not least, a global average pooling layer converts each feature map to a single value by averaging. The output of both branches is concatenated and a fully connected layer with softmax activation to output class probabilities. The cross-validation is necessary to validate the model performance. In this study, StratifiedKFold from scikit-learn library[16] is used to split the training data into training and validation sets for cross-validation. StratifiedKFold is a variant of k-fold cross-validation that ensures each fold maintains the same proportion of class labels as the original dataset. In this study, StratifiedKFold is initialized with 5 splits. The training and validation processes are repeated for each fold, ensuring balanced class distribution in both training and validation sets. After training on each fold, predictions are made on the corresponding test set, and the results are aggregated to evaluate overall model performance. Furthermore, The LSTM-FCN model is trained for 20 epochs After cross-validation, the result of evaluating the LSTM-FCN is shown in figure 5. Here is the explanation of the result:

- precision: Precision is the ratio of correctly predicted positive samples to the total predicted positives. High precision indicates that the model has a low false positive rate.
- recall: Recall is the ratio of correctly predicted positive samples to all samples in the actual class. High recall indicates that the model has a low false negative rate.

```
LSTM-FCN Test Classification Report:
                             precision    recall  f1-score   support

    family_high,stranger_high       0.97      1.00      0.99        36
      family_low,stranger_low       0.80      0.33      0.47        36
   family_low,stranger_medium       0.65      0.92      0.76        36
  family_medium,stranger_high       0.97      0.97      0.97        36
   family_medium,stranger_low       0.86      1.00      0.92        36
family_medium,stranger_medium       1.00      0.97      0.99        36

                     accuracy                           0.87       216
                    macro avg       0.87      0.87      0.85       216
                 weighted avg       0.87      0.87      0.85       216

LSTM-FCN Test Confusion Matrix:
[[36  0  0  0  0  0]
 [ 1 12 16  1  6  0]
 [ 0  3 33  0  0  0]
 [ 0  0  1 35  0  0]
 [ 0  0  0  0 36  0]
 [ 0  0  1  0  0 35]]
```

Fig. 5. LSTM-FCN Test Classification Report and Confusion Matrix

- f1-score: The F1-score is the harmonic mean of precision and recall, providing a balance between the two. A high F1-score indicates both high precision and high recall.
- support: Support is the number of actual occurrences of the class in the dataset. It indicates how many instances of each class are present.
- accuracy: Overall accuracy is the ratio of correctly predicted instances to the total instances. In this case, the accuracy is 0.87, indicating that the model correctly classified 87% of the instances.
- macro avg: Macro average computes the metric independently for each class and then takes the average, treating all classes equally.
- weighted avg: Weighted average takes into account the support (number of instances) of each class when computing the average. This gives a more global measure of performance that accounts for class imbalance.
- The confusion matrix provides a detailed breakdown of the model's performance in terms of actual vs. predicted classifications

As explained above, the accuracy that LSTM-FCN model achieved is 87%. The training and validation loss diagram can be seen in figure 6 and the accuracy diagram can be seen in figure

6.2.2 RF. The data preprocessing is the same as for the LSTM-FCN model. However, the RF model receives a 2D array as input, where each row represents a sample, and each column represents a feature. This contrasts with the LSTM model, which can handle 3D input arrays suitable for sequential data. RF is an ensemble learning method that constructs multiple decision trees and combines their outputs for classification. The parameter grid definition is as follows: param_grid = 'n_estimators': [50, 100, 200], 'max_depth': [None, 10, 20, 30], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4]. And by using GridSearchCV(scikit-learn library[16]) for hyperparameter tuning with 3-fold cross-validation, it ensures that the

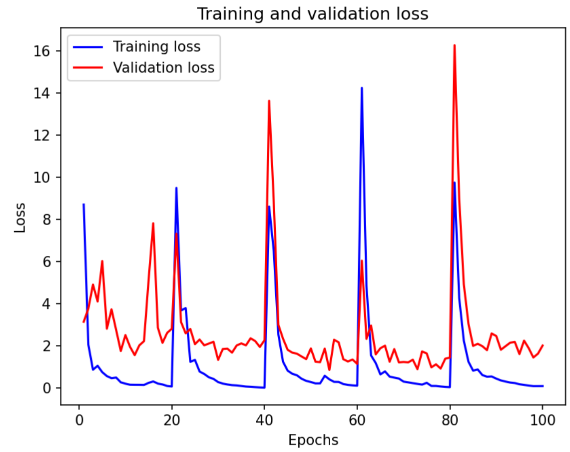


Fig. 6. The training and validation loss

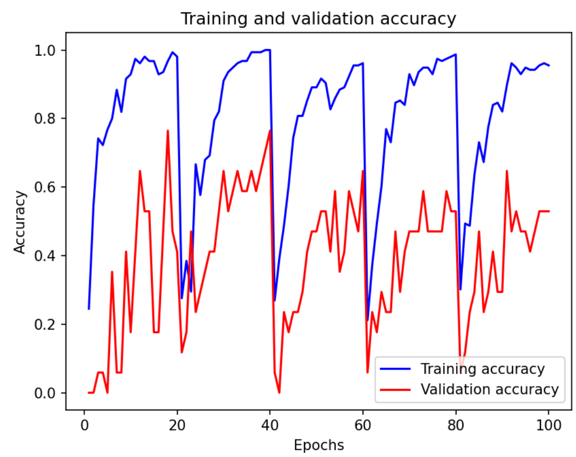


Fig. 7. The training and validation accuracy

best parameters are selected based on validation performance. In this research, the best Random Forest parameters are: 'max_depth': None, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 100. With this setup, the result is shown in figure 8. The detailed explanation of the report can be found in section 6.2.1. The ROC curve is shown in figure 9. The ROC (Receiver Operating Characteristic) curve illustrates the performance of the Random Forest classifier by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold settings. The curve shown indicates that the classifier achieves a high TPR with a relatively low FPR, which suggests that the model has a good ability to distinguish between the positive and negative classes. The area under the curve (AUC) would provide a single metric summarizing the overall performance, with a value closer to 1 indicating a better model(Tan, 2009)[22]. The diagonal line represents a random classifier, so the fact that our ROC curve is well above this line demonstrates that the Random Forest

```
                              precision    recall  f1-score   support

   family_high,stranger_high       0.50      1.00      0.67         1
     family_low,stranger_low       0.83      0.91      0.87        11
  family_low,stranger_medium       1.00      0.40      0.57         5
  family_medium,stranger_high      0.00      0.00      0.00         1
   family_medium,stranger_low      0.50      1.00      0.67         3
family_medium,stranger_medium      0.00      0.00      0.00         1

                    accuracy                           0.73        22
                   macro avg       0.47      0.55      0.46        22
                weighted avg       0.73      0.73      0.69        22

Random Forest Test Confusion Matrix:
[[ 1  0  0  0  0  0]
 [ 0 10  0  0  1  0]
 [ 0  2  2  0  1  0]
 [ 1  0  0  0  0  0]
 [ 0  0  0  0  3  0]
 [ 0  0  0  0  1  0]]
```

Fig. 8. RF Test Classification Report and Confusion Matrix

classifier performs significantly better than random guessing. In conclusion, the model correctly classified 73% of the total instances.

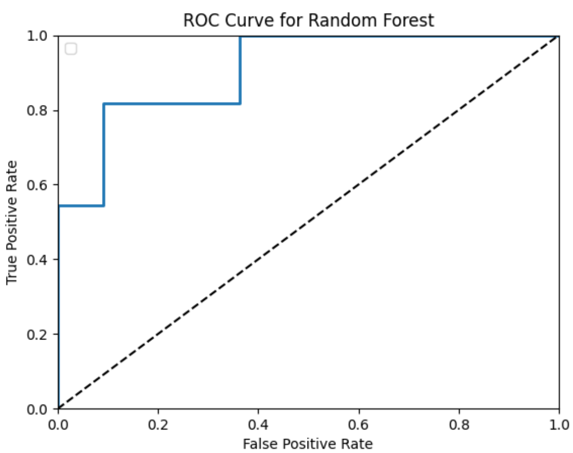


Fig. 9. ROC curve for RF

6.2.3 SVM. This research employs a Support Vector Machine (SVM) classifier to classify the input data. It utilizes PCA (Principal Component Analysis)(Abdi & Williams, 2010)[2] for dimensionality reduction and GridSearchCV(scikit-learn library[16]) for hyperparameter tuning with cross-validation. The training process is very similar to the RF mentioned above. A notable point is PCA, which is applied to reduce the dimensionality of the input data. It transforms the high-dimensional data into a lower-dimensional space while retaining most of the variance. Furthermore, this research uses 5-fold cross-validation for SVM training. The best SVM parameter found after 5-fold cross-validation is 'C': 10, 'gamma': 0.001, 'kernel': 'rbf'.

However, the overall performance of SVM is not well. As shown in figure 10: The detailed explanation of the report can be found in

```
SVM Test Classification Report:
                              precision    recall  f1-score   support

   family_high,stranger_high       0.00      0.00      0.00         1
     family_low,stranger_low       0.50      1.00      0.67        11
  family_low,stranger_medium       0.00      0.00      0.00         5
  family_medium,stranger_high      0.00      0.00      0.00         1
   family_medium,stranger_low      0.00      0.00      0.00         3
family_medium,stranger_medium      0.00      0.00      0.00         1

                    accuracy                           0.50        22
                   macro avg       0.08      0.17      0.11        22
                weighted avg       0.25      0.50      0.33        22

SVM Test Confusion Matrix:
[[ 0  1  0  0  0  0]
 [ 0 11  0  0  0  0]
 [ 0  5  0  0  0  0]
 [ 0  1  0  0  0  0]
 [ 0  3  0  0  0  0]
 [ 0  1  0  0  0  0]]
```

Fig. 10. SVM Test Classification Report and Confusion Matrix

section 6.2.1. Moreover, the ROC curve is shown in figure 11. The

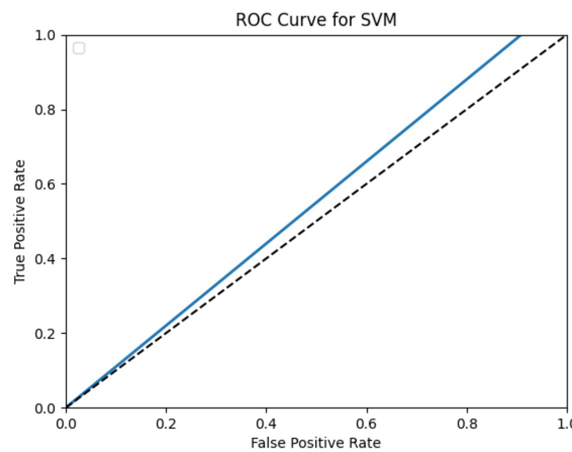


Fig. 11. ROC curve for SVM

overall accuracy for SVM is only 50%.

7 DISCUSSION

7.1 Evaluation

Reflecting on the entire research, the insufficiency of the sample size is one of the reasons why some models did not produce the desired results. Although this issue was unavoidable due to time constraints. In terms of data preprocessing, the researcher initially made an incorrect classification by using the country reported by participants as the label for the touch signals. After discussions with the supervisor, a more reasonable classification method was adopted, involving techniques like K-means clustering.

Regarding the models used, many excellent algorithms and models were not considered in this study. Due to time limitations, experiments were conducted on only the three models mentioned in the paper. The results obtained were: LSTM-FCN achieved an accuracy of 87%, RF achieved an accuracy of 73%, and SVM achieved an accuracy of only 50%. This study used accuracy as the sole metric to compare the models. Clearly, the performance of the LSTM-FCN model was the most outstanding.

This answers the research question: "How can data on social touches performed by people from different cultural backgrounds under different attention-seeking scenarios, collected through Touch Sensitive Patch, be trained by a machine learning algorithm to identify differences between social touches?" The conclusion is that among the studied models, LSTM-FCN is the most ideal method for classifying social touches performed by people from different cultural backgrounds. This involves data preprocessing, such as using K-means to classify individuals and employing SMOTE to enhance the training set, as well as cross-validation for different models. Overall, the experiment yielded a reliable conclusion.

7.2 Future work

It is very unfortunate that this study did not have more time to conduct further research on future applications. However, this study believes that learning and applying unknown social touches on humans, based on correctly classified social touches, is still important. For instance, in nursing homes, companion robots could learn the social touch habits of the individuals they care for, accurately label them, and then perform the same types of social touches to make the individuals feel comforted, safe, and as if they were accompanied by real people.

8 CONCLUSION

This study conducted thorough research on the proposed research question. First, the researchers conducted a comprehensive review of the literature on the relationship between social touch and cultural background and found relevant studies on using machine learning for classifiers. To address the research question, three sub-research questions were posed and answered step by step. Initially, the researchers collaborated with a fellow student to collect data using a touch-sensitive patch. Subsequently, after discussions with the supervisor and independent research, an effective data preprocessing method and sample classification method were developed. Finally, after studying a large body of literature on machine learning for time series classification, the three most suitable models were selected, trained, and tested. The conclusion was that Long Short Term Memory-Fully Convolutional Networks (LSTM-FCN) is the most appropriate machine learning model for this study.

ACKNOWLEDGMENTS

This research was greatly supported by Professor Angelika Mader and Gwenn Englebienne, whose guidance and insights have been invaluable to my study. I am deeply grateful for their assistance in my research on social touch, and for inspiring me to pursue this area of study. Additionally, I would like to extend my heartfelt thanks to my fellow researchers, Daan Schram, and Adrian Josan, for their collaboration in developing the application used for data collection, and for their support in gathering sufficient data and providing relevant assistance throughout the project.

REFERENCES

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. https://www.tensorflow.org/ Software available from tensorflow.org.

[2] Hervé Abdi and Lynne J. Williams. 2010. Principal component analysis. *Wiley interdisciplinary reviews. Computational statistics* 2, 4 (7 2010), 433–459. https://doi.org/10.1002/wics.101

[3] Abien Fred Agarap. 2018. Deep Learning using Rectified Linear Units (ReLU). *arXiv (Cornell University)* (1 2018). https://doi.org/10.48550/arxiv.1803.08375

[4] Carissa J. Cascio, David Moore, and Francis McGlone. 2019. Social touch and human development. *Developmental cognitive neuroscience* 35 (2 2019), 5–11. https://doi.org/10.1016/j.dcn.2018.04.009

[5] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2002. SMOTE: Synthetic Minority Over-sampling technique. *Journal of artificial intelligence research/The journal of artificial intelligence research* 16 (6 2002), 321–357. https://doi.org/10.1613/jair.953

[6] Osisanwo FY, Akinsola JET, Awodele O, Hinmikaiye J. O, Olakanmi O, and Akinjobi J. 2017. Supervised Machine Learning Algorithms: Classification and comparison. *International journal of computer trends and technology* 48, 3 (6 2017), 128–138. https://doi.org/10.14445/22312803/ijctt-v48p126

[7] Alberto Gallace and Charles Spence. 2010. The science of interpersonal touch: An overview. *Neuroscience biobehavioral reviews/Neuroscience and biobehavioral reviews* 34, 2 (2 2010), 246–259. https://doi.org/10.1016/j.neubiorev.2008.10.004

[8] Simone Gumtau. 2005. Communication and interaction using touch-examine the user before you reproduce his hand! *COGNITIVE SCIENCE RESEARCH PAPER-UNIVERSITY OF SUSSEX CSRP* 576 (2005), 18.

[9] Antal Haans and Wijnand IJsselsteijn. 2005. Mediated social touch: a review of current research and future directions. *Virtual reality* 9, 2-3 (12 2005), 149–159. https://doi.org/10.1007/s10055-005-0014-2

[10] Fazle Karim, Somshubra Majumdar, and Houshang Darabi. 2019. Insights into LSTM Fully Convolutional Networks for Time Series Classification. *IEEE access* 7 (1 2019), 67718–67725. https://doi.org/10.1109/access.2019.2916828

[11] Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Shun Chen. 2018. LSTM Fully Convolutional networks for Time series classification. *IEEE access* 6 (1 2018), 1662–1669. https://doi.org/10.1109/access.2017.2779939

[12] V. Kecman. 2005. *Support Vector Machines – An Introduction.* 1–47 pages. 1 https://doi.org/10.1007/10984697\{_

[13] Guillaume Lemaître, Fernando Nogueira, and Christos K. Aridas. 2017. Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *Journal of Machine Learning Research* 18, 17 (2017), 1–5. http://jmlr.org/papers/v18/16-365.html

[14] Rensis Likert. 1932. A Technique for the Measurement of Attitudes. *Archives of Psychology* 140 (1932), 1–55.

[15] Aakash Parmar, Rakesh Katariya, and Vatsal Patel. 2018. A review on Random Forest: An ensemble classifier. *Lecture notes on data engineering and communications technologies* (12 2018), 758–763. 86 https://doi.org/10.1007/978-3-030-03146-6\{_

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[17] A. Saarinen, V. Harjunen, I. Jasinskaja-Lahti, I. P. JäAskëlainen, and N. Ravaja. 2021. Social touch experience in different contexts: A review. *Neuroscience & Biobehavioral Reviews* 131 (2021), 360–372. https://doi.org/10.1016/j.neubiorev.2021.09.027

[18] A. Schirmer, C. Cham, Z. Zhao, and I. Croy. 2022. What makes touch comfortable? An examination of touch giving and receiving in two cultures. *Personality & Social Psychology Bulletin* 49, 9 (2022), 1392–1407. https://doi.org/10.1177/01461672221105966

[19] Alex Sherstinsky. 2020. Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. *Physica. D, Nonlinear phenomena* 404 (3 2020), 132306. https://doi.org/10.1016/j.physd.2019.132306

[20] Ravindra Singh and Naurang Singh Mangat. 1996. *Stratified sampling.* 102–144 pages. 5 https://doi.org/10.1007/978-94-017-1404-4\{_

[21] J. T. Suvilehto, L. Nummenmaa, T. Harada, R. I. M. Dunbar, R. Hari, R. Turner, N. Sadato, and R. Kitada. 2019. Cross-cultural similarity in relationship-specific social touching. *Proceedings of the Royal Society B: Biological Sciences* 286, 1901 (2019), 20190467. https://doi.org/10.1098/rspb.2019.0467

[22] Pang-Ning Tan. 2009. *Receiver operating characteristic.* 2349–2352 pages. 569 https://doi.org/10.1007/978-0-387-39940-9\{_

[23] Siewart van Wingerden, Tobias J. Uebbing, Merel M. Jung, and Mannes Poel. 2014. A Neural Network Based Approach to Social Touch Classification. In *Proceedings of the 2014 Workshop on Emotion Representation and Modelling in Human-Computer-Interaction-Systems* (Istanbul, Turkey) *(ERM4HCI '14)*. Association for Computing Machinery, New York, NY, USA, 7–12. https://doi.org/10.1145/2668056.2668060

[24] Junjie Wu. 2012. *Advances in K-Means Clustering: a data mining thinking.* http://ci.nii.ac.jp/ncid/BB10645375

A APPENDIX: USE OF AI

During the preparation of this work, the author used OpenAI's ChatGPT 3.5 in order to generate the four scenarios from the user study of this research. This helped to formulate nuanced and clear scenarios. After using this tool/service, the author reviewed and edited the content as needed and took full responsibility for the content of the work. The author edited the last lines of the retrieved scenarios to clearly state the goal of the touch.

Furthermore, The author used ChatGPT-4o & DeepL to translate and ChatGPT-4o & Grammarly to refine this article, and consulted ChatGPT-4o on a few questions related to explaining machine learning models. This was done to better convey the intended message to the readers. After using this tool/service, the author reviewed and edited the content as needed and took full responsibility for the content of the work.