Computers, Materials & Continua
DOI:10.32604/cmc.2021.015612
Article

Tech Science Press

Multi-Modal Data Analysis Based Game Player Experience Modeling Using LSTM-DNN

Sehar Shahzad Farooq[1], Mustansar Fiaz[1], Irfan Mehmood[2], Ali Kashif Bashir[3], Raheel Nawaz[4], KyungJoong Kim[5] and Soon Ki Jung[1,*]

[1]School of Computer Science and Engineering, Kyungpook National University, Daegu, Korea
[2]Faculty of Engineering & Informatics, School of Media, Design and Technology, University of Bradford, Bradford, BD7 1DP, UK
[3]Department of Computing and Mathematics, Manchester Metropolitan University, Manchester, UK
[4]Department of Operations, Technology, Events and Hospitality Management, Manchester Metropolitan University, UK
[5]Institute of Integrated Technology, Gwangju Institute of Science and Technology, Korea
[*]Corresponding Author: Soon Ki Jung. Email: skjung@knu.ac.kr

Received: 30 November 2020; Accepted: 23 March 2021

Abstract: Game player modeling is a paradigm of computational models to exploit players' behavior and experience using game and player analytics. Player modeling refers to descriptions of players based on frameworks of data derived from the interaction of a player's behavior within the game as well as the player's experience with the game. Player behavior focuses on dynamic and static information gathered at the time of gameplay. Player experience concerns the association of the human player during gameplay, which is based on cognitive and affective physiological measurements collected from sensors mounted on the player's body or in the player's surroundings. In this paper, player experience modeling is studied based on the board puzzle game "Candy Crush Saga" using cognitive data of players accessed by physiological and peripheral devices. Long Short-Term Memory-based Deep Neural Network (LSTM-DNN) is used to predict players' effective states in terms of valence, arousal, dominance, and liking by employing the concept of transfer learning. Transfer learning focuses on gaining knowledge while solving one problem and using the same knowledge to solve different but related problems. The homogeneous transfer learning approach has not been implemented in the game domain before, and this novel study opens a new research area for the game industry where the main challenge is predicting the significance of innovative games for entertainment and players' engagement. Relevant not only from a player's point of view, it is also a benchmark study for game developers who have been facing problems of "cold start" for innovative games that strengthen the game industrial economy.

Keywords: Game player modeling; experience modeling; player analytics; deep learning; LSTM; game play data; Candy Crush Saga

1 Introduction

Game player modeling is the study of computational models to gain an abstract description of players in various games [1]. This description helps to detect, predict, and express the behavior and experience of players and personalize games according to user preference. These models can be created automatically using computational and artificial intelligence techniques, which are often enhanced based on theories derived from human interaction with games. It offers two major benefits. First, it helps in content customization to cover a broader range of players with different skill levels and adapt challenges on the fly in response to players' actions. Second, it works as a form of feedback for game developers and designers so that they may add innovative features to games as well as develop new games that advance knowledge, synthesize experience, and escalate the interest of players [2].

Experience modeling is the association of a human player during gameplay. Player experience defines the individual and personal experience of a player while he or she is playing a game. It describes the interaction of distinct people with particular game states investigated during or after gameplay. It is based on cognitive and affective physiological measurements collected from sensors mounted on the player's body or in the player's immediate environment [2]. A simple terminology that is used for experience modeling is how a player feels during a game. Player experience modeling is generally considered as learning the subjective nature of a game player during a game. This learning is dependent on an investigation of the player's physiological data to estimate the player's experience in-game. The purpose of this learning is to understand situations within a game that really affect the player's cognitive focus. It helps to analyze the influence of games on users and can be used to generalize the contents of games based on analysis. Furthermore, it can then be used for content generation, content customization, and game customization. A player's experience may be inferred by analyzing interaction patterns during game play and emotions associated with the game [3]. Experience modeling is mapped to data collected during the game play interaction of the player and the player's affective states. The systematic computational analysis of players' physiological data to detect, predict, and estimate the experience of game players is called player analytics. The input to the model includes affective stimuli, emotional responses, and nervous system activities. The outputs of the model are emotional states, personality traits, annotations, rankings, and so on. These outputs are based on the objectives of experiments used for experience modeling. For emotions, the valence–arousal scale by Russell and Plutchik's emotion wheel are the most popular methods of analysis. However, the Big-Five Model and PERSONAGE are used for physiological factor classifications and speech psycholinguistic analysis, respectively [1].

One of the main challenges in the game industry is to collect in-game data on game players. Several companies have recorded data and later used this data for analysis of players' behavior and experience, but it is very costly and usually not made available to the public. Experience modeling, on the other hand, is dependent on players' physiological manifestations and somewhat does not depend on in-game data. This data collection is done in this research and analyzed using the LSTM-based deep neural network approach to predict the experience of game players while playing the "Candy Crush Saga" game employing the concept of transfer learning [4].

2 Literature Review

Game player modeling has been studied vastly in game research and game development [5], where researchers followed standardized approaches of humanities and social sciences for game player modeling. The hypothesis of modeling game players considers the emotions of humans attached to game play.

The traditional models found extensively in the last decade of literature are known as Malone's core design dimensions for behavior modeling using in-game data analysis [6]. It contributes to "fun" in games [7]. In particular, fun is defined in three categories:

- Challenge: For any fun game, a challenge is defined in terms of target achievement, level completion, treasure collection, killing opponents, or defending against an enemy.
- Fantasy: Artificial environment created by game developers to provide entertaining and desirable situations or scenes attractive to game players.
- Curiosity: Expectation of a game player about game states and a player's feeling about the next possible incoming challenge or situation.

All three categories have been studied in car racing games, preschooler games, physical games, and prey–predator games [7]. Furthermore, emotional modeling is articulated by directly mapping emotions into cognitive appraisal theories. One such example is the representation of emotions on the arousal–valence plane. According to this mapping, it is assumed that the increase in the heart rate of game players is directly proportional to arousal. By increasing the heart rate, arousal increases, and high arousal is an excitement state of game players. Other examples include Russell's circumflex model of affect [8] and Frome's comprehensive model of emotional response studies in single-player games [9].

Player modeling or player profile is also studied in the form of Bartle's classification, which has distinguished several types of game players [10]. It defines four standards of classifications (i.e., killers, achievers, socializers, and explorers). In addition, some other methodologies are followed by researchers to exploit specific player experiences [11]. These models follow exact science theories by observing and analyzing the collected data of game players. These models do not work on the hypothesis of game theories; instead, they predict players' states by extracting knowledge from data captured at the time of game play or after game play. For the sake of ease, these algorithms and techniques have been sub-divided into conventional AI methods and deep learning-based methods. An examination of recent research considering the two different categories is discussed below.

2.1 ConventionalHandcrafted AI Methods

To enrich game design, player experience is studied by Koster and lazzaro to modernize the fun and magic in games [1]. Another incorporation theoretical method [12] focuses on players' involvement within games [13]. The designed machine learning algorithm is a combination of Markov's model, state aggregation, and player heuristic search focusing on managing huge data. The purpose is to reduce efforts to extract system-specific features required for a hand-designed model. Puzzle games have been considered to aid in the design of a framework for the movement prediction of game players [14]. A neural network-based approach was developed in [15] by considering game-level parameters (i.e., in-game data), characteristics of players (i.e., behavior), and emotional annotations (i.e., self-reported emotions). The purpose is to predict the action-related experience of gamers during game play. This is done by using information about player style and developing a model to predict gamer preferences during a game. The assumption in this study is that for a particular session, preferences of users remain fixed. However, this does not work for all games and all players. There is a large variation in players' style of dealing with one fixed situation in different ways, which opens room to improve the adaptable model to express the behavior change of players over time. This issue is solved in [16], who proposed to predict dynamic states of game play by capturing fluctuations in behavior. They reported that sequential approaches perform more accurately than non-sequential algorithms with the fact that segments of strategies adopted by game players change over time and situation.

A player's performance is modeled to learn individual player behavior in [17] by developing a tensor factorization technique. The tensor factorization method is capable of adopting alterations in players' game skills. Along with classification algorithms, various regression algorithms are also coupled to clustering methodologies by researchers [18]. The Tomb Raider: Underworld game is examined by [19] to predict different but unique aspects of game players' behavior with the help of supervised learning. Particularly, they predict churning and survival analysis of game players. The linear regression efficacy is demonstrated by considering game play data derived from the first and second levels of Tomb Raider: Underworld. Considering the performance of machine learning algorithms, it is concluded that there are several techniques that can outperform the model experience and behavior of gamers. However, there are several challenges that limit opting for these techniques for game and player analytics. Two of the main challenges are data collection and feature engineering pre-processing. Along with these, computational cost and time consumption are dominant over prediction accuracy.

2.2 Deep Learning-Based Methods

Evolutionary algorithms are the latest approaches for learning player behavior and experience [2]; for instance, genetic algorithms learn gamers' strategies and reinforce the learning of separate expertise levels [20]. The goal-based actions in a sequence provide sufficient explanatory values. These goals can then be used to predict a player's behavior if ascertained (i.e., the gamer must execute these goals in ascending order to achieve the target). Player experience modeling considers the goals achieved by players during game play to predict the effectiveness of the game in relation to gamers. Along with goals, players' actions are also important, as adopted by [21] and several others. The Markov logic network has been adapted to develop a successful goal recognition framework for players [22]. It employs model parameters obtained from player interactions in non-linear games to enable an automatic goal recognition approach. Similarly, problem-solving goals are linked to game events using probabilistic inference and logical reasoning to model players [23]. When empirically tested and compared, it was found that models discovering game events surpassed all state-of art previous best approaches. Another input–output hidden Markov model method was proposed in [24] to predict player goals in action games. Initially, there was no significant performance difference between the model that was trained for a first-time player and the model trained by the experimenter. However, subsequent models were improved with time. The conclusion is that the target achievement system performs well for players who have developed game playing styles rather than general players. Although this approach had shown a challenging performance over hand-authored finite state machines, it could not breach deep learning approaches [25]. A goal recognition framework is based on stacked de-noising auto-encoders using a deep learning algorithm, trained through game player interaction, without feature engineering. The deep neural network approach was adapted in [26], where researchers asked players to play procedurally-generated levels of Super Mario and collected data from more than a hundred players. The network outperformed the predicted emotional features with this data. Several other neural networks, convolution neural networks, recurrent neural networks, and LSTM-based techniques have been employed to predict gamers' experience and behavior [1]. Also, several domain-to-domain transfer learning approaches in many video games [27] have been employed; however, there remain some unexplored areas such as transfer learning for different domains.

Transfer learning focuses on gaining knowledge while solving one problem and using the same knowledge to solve a different but related problem [4,28]. For example, training a model with categories of book A and testing it on book B with different categories [4], Cognitive learning on different Atari games, and constraint-based generalization [29]. Instead of different data distributions, there can be situations where the features of one dataset are different from the features of another dataset or where labels that have been used during training are different in testing or missing altogether, as in unsupervised learning [30]. Such problems can be solved using transfer learning methods with the fact that the performance of these methods will be degraded.

The concept of transfer learning arises when knowledge of one domain is adjusted to use it in another domain [31]. The challenge is to consider the beneficial knowledge that can be shared between the two different domains. However, when the target of the prediction model is similar in both domains, the solution to the problem becomes very easy. The proposed game player modeling framework in this study is based on LSTM-DNN with the concept of transfer learning of knowledge obtained in one domain and used for prediction in another domain. The experience of game players in playing video games is predicted by training an LSTM-DNN model with the experience of subjects while watching video games. In both cases, although emotional states are used as a matter of experience for both video watching audiences and video game players, this study opens a new arena for game developers that face problems of "cold start" when designing a new game. The solution to this problem is two-fold. First, this research overcomes the challenges of collecting physiological datasets for game players playing different video games. Researchers can train their models to some other physiological-based datasets that are publicly available over different domains and can test these models on video games. Secondly, although the video game community has a healthy feedback mechanism to get immediate feedback through demos, beta testing, or within the community, it has been noticed that game developers who launch a new video game usually wait for months to get reality-based feedback about the new game in order to improve the content and targets of the game. However, during that time, most audiences who have waning interest in a game leave with the intention that they will not return to the game again, as it does not maintain their interest and reduces their motivation to play it. The proposed method facilitates researchers' desire to train models with other subjects' experiences, and they can test newly launched games without waiting for feedback. This methodology can speed up game design, target development, and encourage novel innovations in game content. In this respect, LSTM-based approaches apparently solve a major problem in game player modeling, at once offering superior accuracy in goal recognition and eliminating the exhaustive laborious feature engineering previously required for such modeling.

3 Proposed Methodology

In this section, we explain the mechanics of the underlying experience modeling framework. The major focus of the proposed framework is PC games, with the intention to accurately predict and model different characteristics of game players, making existing and coming games more realistic, fun, and fruitful for the gaming community. In this case, we employed the power of deep learning and determined that LSTM deep neural networks are the most suitable neural network architecture for exploring the underlying problems outlined above. The proposed framework is a combination of two different cases in terms of datasets, where each dataset considers a variety of different features, both in numbers and physiognomies, to explore and study in depth all possibilities of molding an accurate profile for different audiences. The experimental modeling is built for the Candy Crush Saga game, considering physiological and peripheral sensor data signals in conjunction with LSTM and DNN. The deep neural network performs well to predict the experience of gamers using the concept of transfer learning with remarkable performance. Further details of the method are given in subsequent sections.

To deal with time series data as an input to the network and to implement the LSTM network layer in the DNN, several other layers are also adjusted along with their parameters. This includes an embedding layer before the LSTM layer, a dense layer after the LSTM layer, an activation layer, a dropout layer, another dense layer, and an output layer with a second activation layer. The overall structure of the LSTM-DNN is shown in Fig. 1. The reason for using an embedding layer before the LSTM layer is that it can find similarities between given data of multi-dimensional space. This allows us to see relationships between data distribution and sequences that can be turned into a vector through an embedding layer. For example, when we convert the sentence "deep learning is very deep" to vector "1 2 3 4 1," the word "deep" is repeated two times, and in its vector, we can see that "1" is two times. Embedding layers find such relationships in data for sequence learning.

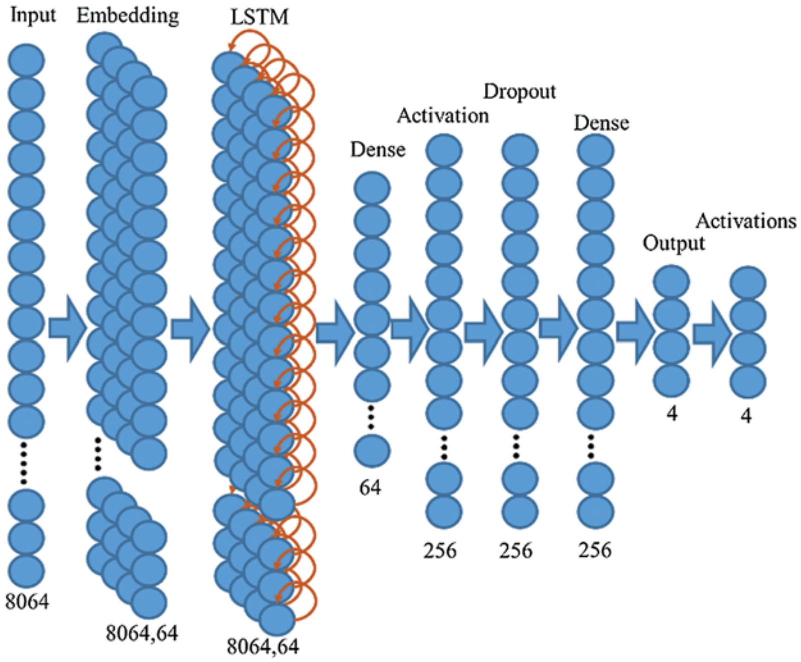


Figure 1: LSTM-DNN with input layer, embedding layer, LSTM layer, dense, and activation layers

The embedding layer is followed by the LSTM layer, which can learn long sequences in data. In the case of the proposed method, where time series data come from physiological and peripheral devices, data of final (last) time depend on data taken at the initial time. Therefore, an LSTM with a size of 64 memory cells can learn long-term sequences in data. It enhances the prediction performance of the model. It has been seen that the proposed method of DNN with an LSTM layer produces good results in terms of accuracy as compared to a simple DNN without an LSTM layer. The input layer shown in Fig. 1 consists of 8064 nodes as an input. The time series data is inputted into the model using these 8064 input nodes. The embedding layer is used to create vectors for the incoming data. It sits between the input and LSTM layers (i.e., the output of the embedding layer is input to the LSTM layer). The weights for the embedding layer can be initialized with random values and can optionally be fine-tuned during training. After the embedding layer, the LSTM layer is used. The role of the LSTM layer is to learn long-term dependencies. The dense layer is a linear multiplication of every element of input. The sigmoid function is used as a non-linear activation function. The dropout layer is used to reduce overfitting by the model. Finally, the output of four labels is predicted, as shown in Fig. 1.

The proposed experience modeling is done in a way that data is collected (explained in a later section) from physiological sensors mounted on the body of the subject while the subject is playing the game. The Emotiv device is mounted on the head of the subject to collect physiological signals from the brain. The IOM Divine is attached to the left-hand fingers of the subject. The left hand of the subject is required to remain still to avoid movement noise during data collection. The right hand of the subject is allowed to control the mouse while the subject is playing the game. The game is shown on a monitor, alongside the subject's EEG signals to keep track of their responses. A Zephyr Bioharness^{tm} 3 belt is worn around the chest, and respiration is measured and saved in the device. Data from the Zephyr belt is then downloaded to provide an input to the model. The collected data is pre-processed, integrated, standardized, and given to the LSTM-DNN for prediction (Fig. 2).

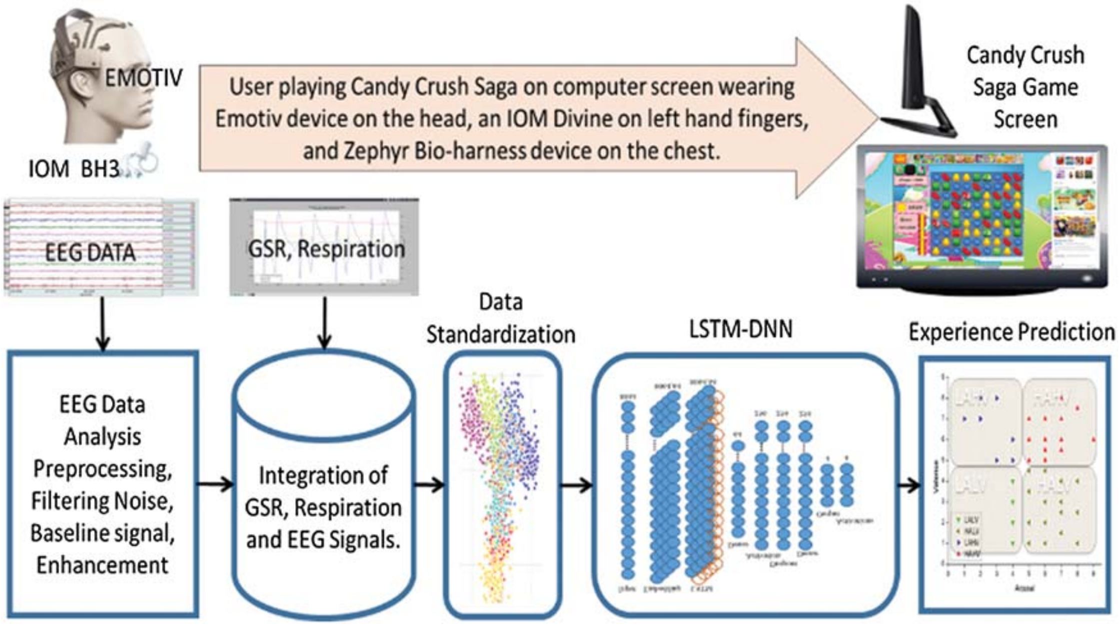

EMOTIV
IOM BH3
User playing Candy Crush Saga on computer screen wearing Emotiv device on the head, an IOM Divine on left hand fingers, and Zephyr Bio-harness device on the chest.
Candy Crush Saga Game Screen
EEG DATA
GSR, Respiration
Data Standardization
LSTM-DNN
Experience Prediction
EEG Data Analysis Preprocessing, Filtering Noise, Baseline signal, Enhancement
Integration of GSR, Respiration and EEG Signals.

Figure 2: An overview of the proposed LSTM-DNN-based experience modeling framework

4 Data Collection and Formation

Emotions are defined as psycho-physiological fluctuations prompted by several external and internal stimuli, such as an object, situation, or contextual environment [32]. Specifically, in terms of music videos, the physiological responses of humans (subjects) are affected by scenes, situations, and context in video clips [33]. The subjects can express their emotions after watching videos. They may also express attractions and distractions during video clips. Along with these, liking/disliking and dominance of video clips are also recognizable. Video games provide the same media as that of video clips, except that the player controls an avatar in the game and performs several actions to achieve a goal. This is somewhat similar to watching video clips with the addition of performing physical activity while playing games. Along with some emotions elicited by watching video game screens, players are more concerned about playing video games and trying to achieve the target. The physiological response of players is, therefore, not only due to watching video games, but also based players' affective response to playing win/lose games and performing in games to achieve targets. The physiological responses of an audience (while watching video clips) and players (while playing video games) can be considered as a dataset of two different domains. Since we have no intention at this time toward the context of video clips and video games, we can say that video clips and video games have the same physiological stimuli, except that the player also focuses on thinking and playing games. In this way, we can use the two datasets of different domains and can determine the similarities and dependencies between them. For this purpose, we used the DEAP dataset (explained in a later section) and collected the Candy Crush Saga dataset to implement the concept of transfer learning using LSTM-DNN.

4.1 Candy Crush Saga Dataset Collection

Candy Crush Saga[1] is the most popular social game played on mobile[2] as well as on desktop computers.[3] It is a puzzle match-3 game [34–36] in which players match candies in combinations of three or more to win points and overcome obstacles or rounds. Upon the successful combination of three candies, the combination vanishes from the board, and some unknown pattern of candies falls from the top, filling the board with a new pattern. In the Candy Crush Saga game, players progress through a colorful candy world after completing the target at each individual level. There are more than 5,300 levels, each of which offers a different puzzle challenge.[4] Each level in Candy Crush Saga has a defined target. There are six colors of candies in Candy Crush Saga[5]: blue, red, orange, yellow, green, and purple. In addition, there are some other special candies that appear on the board if the player combines more than three candies at a time. Some of the most popular names of special candies are "striped," "wrapped," "color bomb," "jelly fish," and "coconut wheel." These special candies are triggered by matching special combinations of four, five, and six candies of the same color together. These candies are sometimes given as a bonus incentive to players for high performance or quick combinations of candies within a specific time.

There are three types of goals found in specific levels: 1) clear all jellies of a particular level with unlimited time and a limited number of moves, 2) achieve a target score in a particular level with limited time and an unlimited number of moves, and 3) clear jellies and achieve a target score with an unlimited number of moves. Based on the above-mentioned challenges, levels are categorized as Easy, Hard, and Super Hard. **Easy Level:** A level in which the player has enough moves to achieve the goal even if the player makes a lot of mistakes. i.e., level 8. **Hard Level:** A level in which the player has limited time to achieve the goal while also permitting the player to make fewer mistakes than they could at the Easy level. i.e., level 125. **Super Hard Level:** A level

[1] https://king.com/game/candycrush.

[2] https://play.google.com/store/apps/details?id=com.king.candycrushsaga&hl=en_US.

[3] https://www.microsoft.com/en-us/store/p/candy-crushsaga/9nblggh18846?activetab=pivot%3aoverviewtab.

[4] http://candycrush.wikia.com/wiki/List_of_Levels.

[5] https://tipsandtricksfor.com/candy-crush-candy-information.htm.

in which the player has both limited time and limited moves and has very little flexibility to make mistakes. i.e., level 343. Super Hard levels may need tricky moves, bonuses, and special candies to achieve specific goals; otherwise, the level cannot be successfully completed. This classifies the game players into novice, average and expert players. Though low average and high average players are also found based on the no. of levels played by them.

Table 1: Definitions of valence, arousal, dominance, and liking in binary scale

Less than 5	5	Greater than 5
Low valence	→	**High valence**
Sad	**Neutral**	Joyful
Unhappy		Happy
Unpleasant		Pleasant
Not interested		Interested
Low arousal	→	**High arousal**
Calm	**Neutral**	Stimulated
Bored		Excited
Deactivated		Activated
Relaxed, easy		Stressed, difficult
Low dominance	→	**High dominance**
No control	**Neutral**	Full control
Does not understand		Understand
Does not make sense		Make sense
Does not involve		Involved
Unlike		**Highly like**
Does not like it	**Neutral**	**Like it**

The emotional definitions of the arousal, valence, and dominance scale are defined with a binary scale. A value of less than five is marked as a low valency (0). Values greater than five are marked as high valency (1). Tab. 1 shows the definitions of emotions for valence, arousal, dominance, and liking. The novel experiment was performed in the Cognition and Intelligence Laboratory in the Department of Computer Science and Engineering, Sejong University, Seoul, South Korea. Fifteen participants aged between 21 and 34 (mean age 27.53) took part in the experiment. Before recording data for the game players, each participant completed a demographic form including name, age, skill level, number of game levels played, and a questionnaire about their physical and mental health. Next, participants were instructed about the experiment, including the task and restrictions which would apply while recording data. While game play requires some physical movement (a hand to control mouse in this case), it was requested that participants keep extraneous movement to a minimum. As a demonstration, the researcher mounted devices and showed them to participants, advising them about the maximum movements they were allowed to collect accurate data. The Candy Crush Saga game was played using a trusted Microsoft app store on the Windows 10 Pro (64-bit) operating system with an Intel® Core™ i7-4790 CPU@3.60 Ghz (8 CPUs), 32 Gigabyte RAM, and an LG ultrawide (HDMI) monitor. On one side was a display of the Candy Crush Saga game, while on the other side was a display of the EEG signal recordings using the Emotiv Control Panel (Fig. 3).

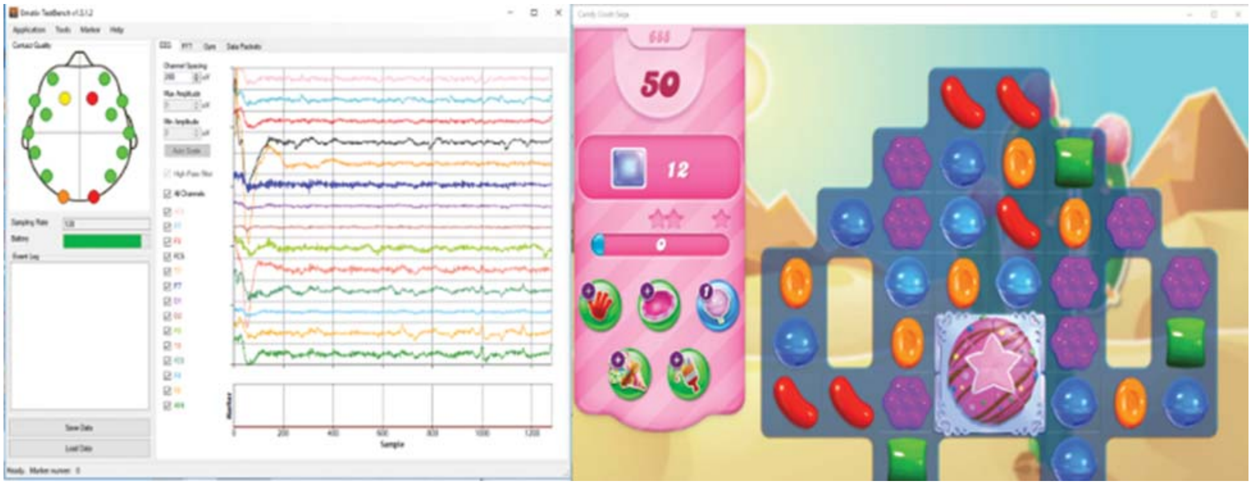


Figure 3: A screenshot of the Candy Crush Saga game and EEG signal recordings

4.1.1 Physiological Data Collection

The EEG physiological signals were recorded using the Emotiv EPOC+[6] (model: Emotiv Premium). The Emotiv EPOC is a pre-made head-mounted device with 14 fixed point electrodes mounted on a wireless headset. These 14 electrodes were fitted to a headset device following the international 10–20 system (Fig. 4 [left]).

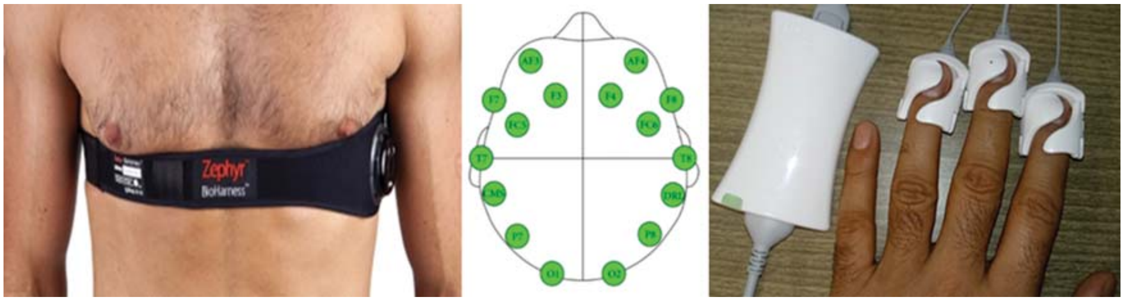


Figure 4: Left: The IOM wild divine grapher connected to subject's left hand, Middle: EEG electrode positions over head, Right: Zypher bioharness divine worn around the chest

The format of the physiological data collected for Candy Crush Saga and its contents are shown in Tab. 2. Several pre-processing steps were taken in order to make the data usable for analysis. The raw data of 14 channels was first down sampled to 128 samples per second, common average referenced, and filtered with a 2 Hz cut-off frequency. In order to maintain resemblance of the Candy Crush Saga data with DEAP data, 63 s of data were finally extracted for each channel. The sequence of EEG signals and peripheral physiological signals is shown in Tab. 3.

[6] https://www.emotiv.com/.

Table 2: Candy Crush dataset (CCD)

Name	Array shape	Contents
Data	40 × 16 × 8064	CCS game levels × Channels × Data
Labels	40 × 4	Game levels × Label (valence, arousal, dominance, liking)

Table 3: Candy Crush data channel description

Channel number	Channel content	Channel number	Channel content
1	AF3	9	P7
2	AF4	10	P8
3	F3	11	FC5
4	F4	12	FC6
5	F7	13	01
6	F8	14	02
7	T7	15	GSR
8	T8	16	Respiration

4.1.2 Galvanic Skin Response (GSR) Data Collection Using IOM Wild Divine

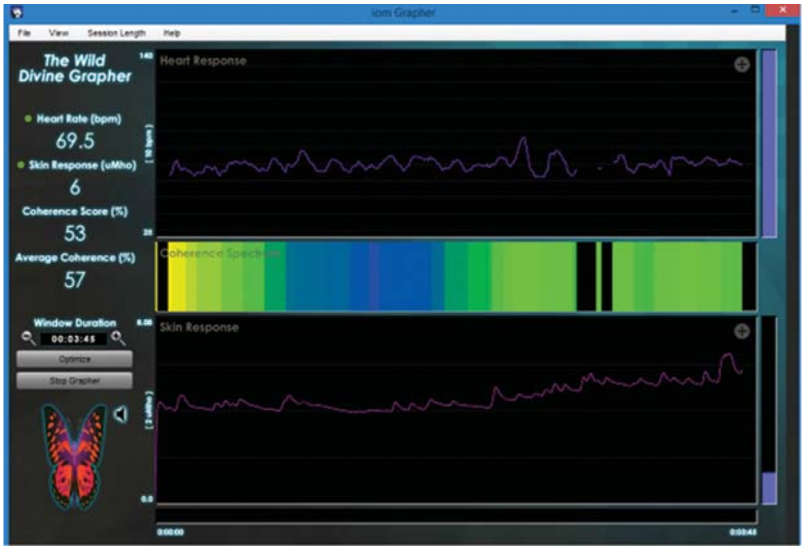
The GSR (skin conductance) of subjects was recorded using the Wild Divine Grapher device and its software on a separate PC with the same specification as described above. The Wild Divine Grapher is a biofeedback system consisting of three sensors worn over the three middle fingers of a hand. The middle sensor measures heart rate, while the remaining ring and index finger sensors measure the skin conductance of the subject. The software displays recorded signals, and data can be stored easily. Figs. 4 (right) and 5 show the Wild Divine Grapher device connected to the subject's left hand and recorded signals in the IOM Grapher software, respectively.

4.1.3 Respiration Data Collection Using Zephyr Bioharness

The respiration data of the subjects were recorded using the Zephyr Bioharness^{tm} 3 belt. This is an advanced physiological monitoring wearable device worn around the chest (Fig. 4 [left]). It provides Bluetooth connectivity to monitor real-time heart rate, breathing rate, accelerometer, posture, skin temperature, respiration, and electrocardiography (ECG). Data are recorded and can be downloaded to the computer for further processing.

4.1.4 Participants' Self-Assessment for the Candy Crush Saga Game

After playing each level of Candy Crush Saga, each player filled a proforma of their emotional state in terms of arousal, valence, dominance, and liking, using a scale from zero to nine, as described earlier. Each subject was asked to express their emotions as accurately as possible. However, it was noticed that some subjects could not understand their emotions while playing the game. Therefore, for a few subjects, the experiment was repeated, and the participants were asked to keep their feelings in mind to represent them accurately at the end of each game play. Since each scale had already been explained to subjects, most of the subjects did not have any issues filling in the proforma and expressed their emotions as they understood them.



Heart Rate (bpm) 69.5

Skin Response (uMho) 6

Coherence Score (%) 53

Average Coherence (%) 57

Window Duration 00:03:45

Figure 5: The IOM grapher software records and displays the heart rate and skin conductance levels of the subject

4.2 DEAP Dataset

The DEAP dataset is a publicly available multi-model database [32]. It consists of a 40-channel dataset: a 32-channel EEG physiological signal and eight peripheral physiological signals. The data of 32 subjects were recorded while each subject was watching a one-minute duration music video clip. Each subject responded to 40 watched videos by expressing their emotions in the form of valence and arousal. The standard Russell's scale of valence and arousal was chosen to represent the emotional affective states of subjects by watching videos. The scale ranges in discrete values from 1 to 9. For arousal, 1 represents a bored state, whereas 9 represents an excited state. Similarly, for valence, 1 represents an emotional state in the form of sadness, while 9 represents happiness. The in-between scales are a combination of arousal and valence charts. Along with valence and arousal, each subject was asked to also scale dominance and liking of the videos. Dominance represents the scale of weak feelings (1) to empowered feelings (9), whereas the liking scale measures participants' interest. The purpose of the DEAP dataset is to analyze the emotional influence of music videos on subjects' states. However, several other emotional predictions, subject classifications, and emotional categories have been studied on this dataset [37–40].

In order to use the DEAP dataset for the experimental evaluation of the model as well as the proposed method, a further pre-process dataset was needed. First, it was necessary to define a classification methodology for the stimuli stated by subjects [41]. We preferred binary classification, as has been suggested in [32]. This was done by thresholding stimuli assessments into half of the defined scale. The value ranges from 1 to the middle of the scale of arousal, valence (i.e., 4.5) were marked as zero, and the remaining high range values were marked as one using a binary classification. Similarly, dominance and liking were also classified as binary. Thus, binary labels were generated for each subject and each video watched by subjects as stated in Tab. 1.

5 Evaluation and Analysis

After collecting game players' data and pre-processing to minimize differences between the DEAP data and the collected data (in terms of sampling rate, dimensions, labels, and format), data were given to LSTM-DNN as an input. Considering variance and performance statistics, data were split into 75%:25% ratios for training and testing. Several experiments were then conducted in order to test the performance of the LSTM-DNN model as well as to validate the homogeneous transfer learning approach to obtain significant results. Since we had different proportions of datasets, we divided the experiments into different categories. A total of seven experiments were conducted using the proposed methodology in order to validate the results of the proposed method. Several other standard machine learning algorithms were also used to compare the results of the proposed method. Finally, state-of-the-art results were compared with the proposed framework results. For classification performance measurement, the most important and most popular method is accuracy calculation, which identifies samples as classified correctly. For binary classification tasks, accuracy is used. Another measure of system performance is the F-measure to summarize the results of each method, considering the fact that there may be imbalanced instances in the binary classification. The F-measure considers precision and recall of the test to compute a score. Precision is the number of correct positive results divided by all positive results predicted by the classifier, and recall is the number of correct positive results divided by all positive samples.

For the experiments, the DEAP dataset was used to train the LSTM-DNN, and the same DEAP dataset was used to test the experience prediction of the LSTM-DNN along with the Candy Crush Saga data collected. In this way, we divided the experiments into seven different types. For the first experiment, all 40 channels of the DEAP dataset were used. Since the collected Candy Crush Saga dataset does not have 40 channels of data, only 25% of the DEAP dataset with 40 channels was used as a testing dataset. For the other experiments, 16 channels were used from the DEAP dataset as well as the Candy Crush Saga dataset. Tab. 4 shows different types of experiments conducted using the LSTM-DNN.

Table 4: Types of experiments conducted using the LSTM-DNN (DD: DEAP dataset, CCD: Candy Crush dataset)

No.	Training	Testing	Name
1	**DD (40)**	**DD (40)**	**LSTM analysis on complete DEAP dataset**
2	DD (16)	DD (16)	LSTM analysis on partial DEAP dataset
3	CCD (16)	CCD (16)	LSTM analysis on Candy Crush dataset
4	**DD (16)**	**CCD (16)**	**LSTM based transfer learning with partial DEAP dataset**
5	**CCD (16)**	**DD (16)**	**LSTM based transfer learning with Candy Crush dataset**
6	DD (16) + CCD (16)	CCD (16)	LSTM based fine-tuned transfer learning with partial DEAP dataset and Candy Crush dataset
7	CCD (16) + DD (16)	DD (16)	LSTM based fine-tuned transfer learning with Candy Crush dataset and partial DEAP dataset

5.1 LSTM Analysis on Complete DEAP Dataset

In this experiment, the DEAP dataset was used to train the LSTM-DNN, and the same data were used to test the experience prediction of the LSTM-DNN. For this purpose, all 40 channels were used. The LSTM-DNN was trained using 75% of the DEAP dataset, and the remaining 25% was used for testing. Fig. 6 shows the performance of the LSTM-DNN overall rating scales (i.e., arousal, valence, dominance, and liking). The best accuracy performance was 71.25% for 60 epochs. As stated, in this experiment, all 40 channels of the DEAP dataset were used, and it can be seen that there were more possibilities of similarities in the channels' data, confusing the LSTM-DNN and reducing its performance for correct instances. To further clarify, it can be said that there are different combinations of each channel over longer-term memory gates that predict zero class in one instance, and the same combinations predict 1 in another instance. The performance of the LSTM-DNN is reduced by overfeeding redundant data. To conclude, it can be said that the LSTM-DNN outperformed in this experiment, and the expected accuracy was more than 80%. However, with so many channels, the possibility of noise and other artifacts is higher than that of a few channels; therefore, the accuracy could not reach the expected level. The F-measure for this experiment was 0.69 (precision 0.68 and recall 0.71).

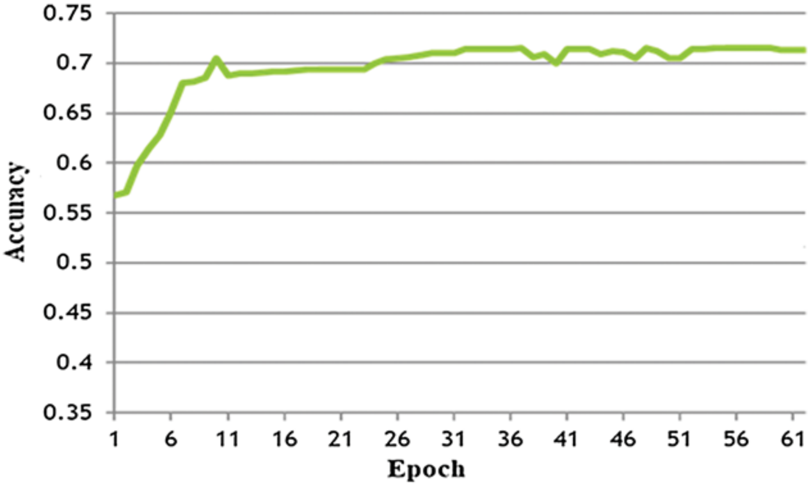


Figure 6: LSTM-DNN performance training: DD [40], testing: DD [40]

5.2 LSTM-DNN Analysis on Partial DEAP Dataset

In the second experiment, the DEAP dataset was used but with a smaller number of channels. By keeping the same experimental conditions as that of experiment one and reducing the DEAP dataset from 40 channels to 16 channels (considering the same channels that are used to collect data from the Emotiv device), the accuracy of the LSTM-DNN was found to be 64.86%. The performance of the LSTM-DNN was reduced by less than 7% from the first experiment, whereas only around one third of channels from the dataset were used in this experiment. In spite of this, the resultant performance still approached the first experiment's results. It can be seen that by reducing the number of channels, redundant data may have been reduced. However, along with it, some informative data is also lost that reduces performance from 71% to 65%. The performance of 60 epochs for this experiment is shown in Fig. 7. The F-measure was 0.64 (precision 0.65 and recall 0.64).

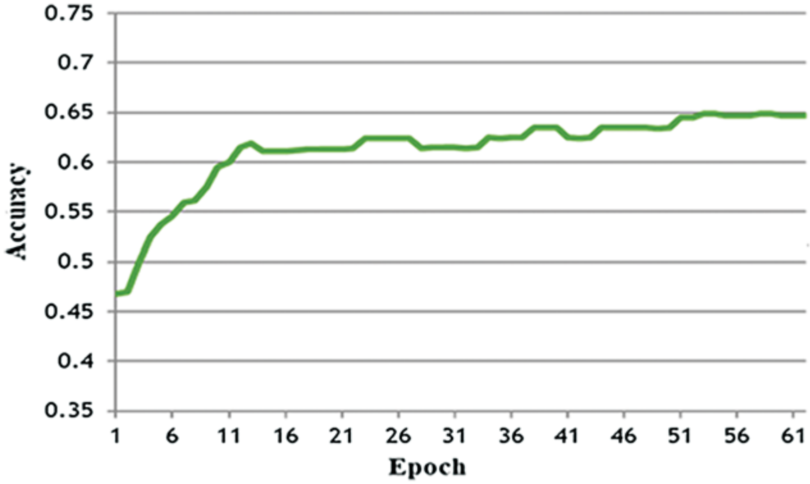


Figure 7: LSTM-DNN performance training: DD [16], testing: DD [16]

5.3 LSTM-DNN Analysis on the Candy Crush Dataset

The LSTM-DNN is validated on the DEAP dataset, which is a publicly available dataset, and several researchers have used that dataset by adopting different machine learning algorithms. However, this experiment was conducted purely on the Candy Crush dataset collected by the author. The Candy Crush data was divided into 75:25 ratios and was used to train the LSTM-DNN. The testing performance was 60%. However, it can be seen that model performance was very smooth compared to other experiments and that the model achieved the highest accuracy with the first 20 epochs. The F-measure for this experiment was 0.59 (precision 0.58 and recall 0.60). Fig. 8 below shows the performance of LSTM-DNN over the Candy Crush dataset.

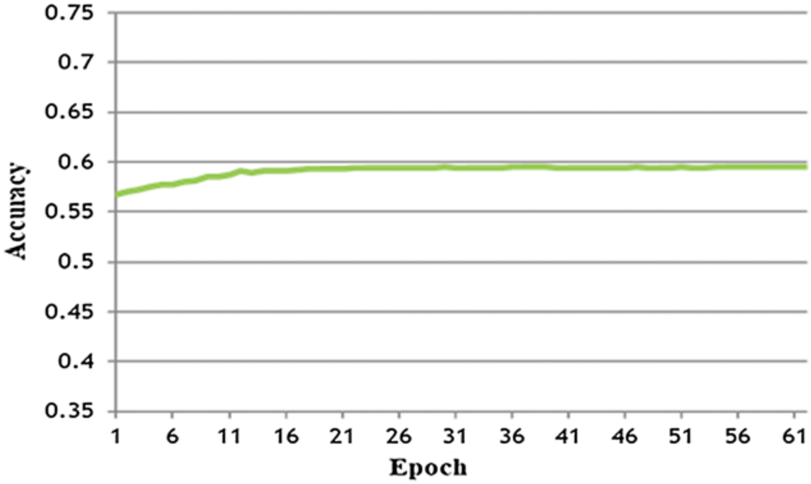


Figure 8: LSTM-DNN performance training: CCD [16], testing: CCD [16]

5.4 LSTM-DNN-Based Transfer Learning with Partial DEAP Dataset

The fourth and most important experiment that was conducted in this study was totally dependent on homogeneous transfer learning. In this experiment, the DEAP dataset with 16 channels (while subjects were watching music video clips) was used to train the LSTM-DNN, whereas testing of the LSTM-DNN was done on the Candy Crush dataset (collected while subjects were playing the game). The purpose of this experiment was to understand the concept of homogeneous transfer learning, which is defined as the learning of a network using one dataset and testing the network on another but similarly structured dataset. The number of channels in both datasets is the same, the frequency of data, and the time length of both datasets is made equal. The performance of the LSTM-DNN was very impressive, with accuracy approaching 62.5%, nearly equal to the second experiment with an accuracy of 64.86%. These are expected results in the homogeneous transfer learning scenario, as the dataset collected while the subjects were playing Candy Crush was different from the DEAP dataset where subjects were just watching music video clips. It can be seen in Fig. 9 that the accuracy graph is prone to noise, especially from the 45th to 50th epochs. The accuracy is reduced, but the network gains accuracy in the last epochs. The final performance was 62.5%. These are the best results that have been found with a homogeneous transfer learning approach. The results may be improved with some amendments to the methodology of the homogeneous transfer learning approach, but that will add computational cost and require more time than this study. The F-measure for this experiment was 0.61 (precision 0.61 and recall 0.62).

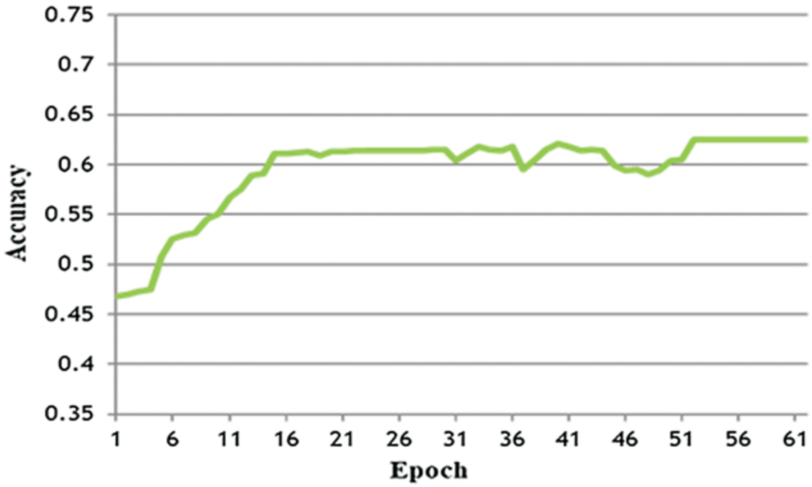


Figure 9: LSTM-DNN performance using transfer learning training: DD [16], testing: CCD [16]

5.5 LSTM-DNN-Based Transfer Learning with Candy Crush Dataset and Partial DEAP Dataset

In this experiment, the LSTM-DNN was trained using Candy Crush data, whereas performance was tested on the DEAP dataset. Since the training data was considerably less than the test data, the LSTM-DNN underfits and does not perform over 50% (Fig. 10). This is true when the data given to the LSTM-DNN do not have many instances that have to learn and predict during testing. The F-measure for this experiment was 0.42 (precision 0.43 and recall 0.41).

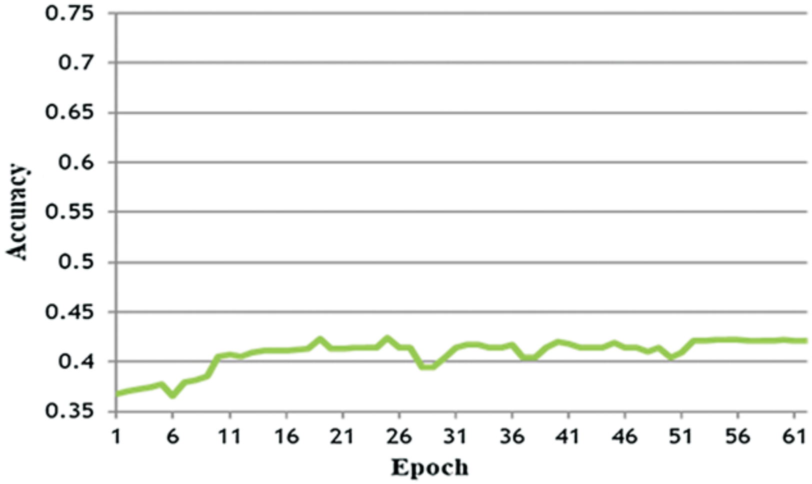


Figure 10: LSTM-DNN performance using transfer learning training: CCD [16], testing: DD [16]

5.6 LSTM-DNN-Based Fine-Tuned Transfer Learning with Partial DEAP Dataset and Candy Crush Dataset

In this experiment, we implemented fine-tuned transfer learning over the previous experiment. Fine-tuning transfer learning refers to training the model substantially on one dataset from one domain and then fine-tuning it with additional training on a second, different domain dataset, rather than testing the fine-tuned model with the dataset of the second domain. This fine-tuned model should achieve more accurate predictions as the weights in the training model are readjusted with the new domain's data. These same results were achieved in this experiment. We trained the model with a partial DEAP dataset with 16 channels for 50 epochs, and then the model was provided with 25% of the Candy Crush dataset to retrain for 10 more epochs. After the training was finished, the trained model was tested with 75% of the Candy Crush dataset. The results of this model can be seen in Fig. 11. It is interesting to note that fine-tuned transfer learning provides better results than simple transfer learning, where we trained the model with one dataset and tested it with a different dataset. However, it has been seen from the experiment that the result is not improved too much (the improvement is just 1% in accuracy). The reason is that when the model was trained with DEAP data for 50 epochs, the weights of the LSTM-DNN were adjusted according to the DEAP data. During the last 10 epochs, with the Candy Crush dataset, the LSTM-DNN tried to readjust its weights according to new data; however, first, based on the Candy Crush dataset, weights changed. Second, due to the short number of training epochs, weights did not adjust accordingly to their best. Even though the final prediction accuracy was increased, the expected performance was not obtained. One solution to increase performance would be to train for more epochs and with more data. However, the limited dataset for the new domain restricts us to a small training dataset. Once weights are readjusted, more epochs may not improve the performance further because of overfitting.

5.7 LSTM-DNN-Based Fine-Tuned Transfer Learning with Candy Crush Dataset and Partial DEAP Dataset

In this experiment, the fine-tuned transfer learning concept was implemented again. The LSTM-DNN was trained with the Candy Crush dataset of 16 channels for 50 epochs and retrained with 25% of the partial DEAP dataset for 10 more epochs. After training was finished, the newly trained model was tested with 75% of the DEAP dataset. The results of this experiment can be seen in Fig. 12. The results of fine-tuned transfer learning for this experiment are highly improved compared to previous experiments. This is because the Candy Crush dataset is very small to input into the model for 50 epochs training. The LSTM-DNN overfits itself by learning almost all the instances in it. However, retraining with the DEAP dataset readjusts its weights to new instances. Since the DEAP data is 40 times that of the Candy Crush dataset, 25% of the DEAP dataset is too big data compared to the Candy Crush dataset, and fine-tuning transfer learning is the same as retraining the LSTM-DNN. Consequently, results are improved instantly. This type of training and transfer learning is not in common practice, as if there is enough data for training the LSTM-DNN in one domain, a pre-trained model may not be needed since the model may train itself quickly even with random initialization.

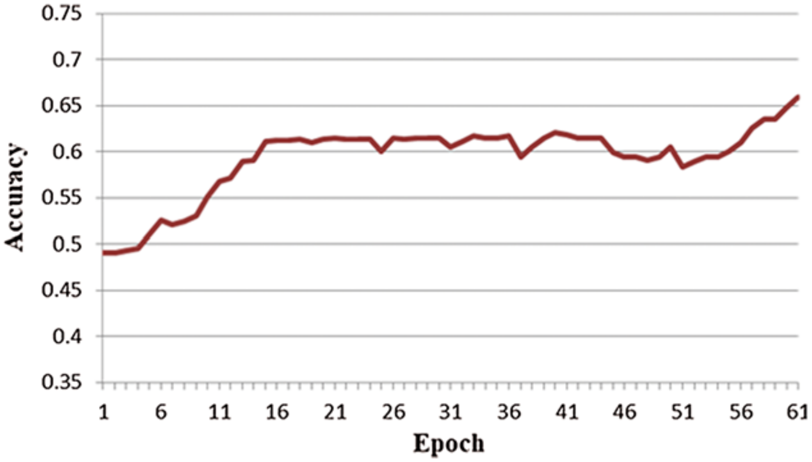


Figure 11: LSTM-DNN performance using fine-tune transfer learning training: DD [16] + CCD [16], testing: CCD [16]

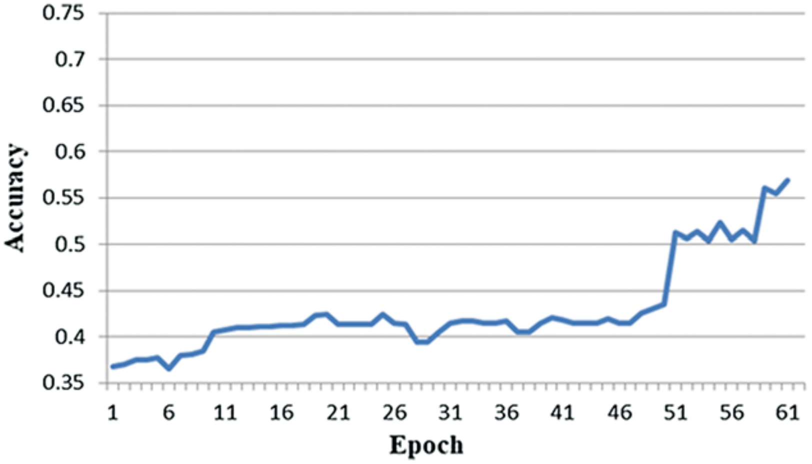


Figure 12: LSTM-DNN performance using fine-tune transfer learning training: CCD [16] + DD [16], testing: DD [16]

Tab. 5 summarizes all experiments conducted using the proposed LSTM-DNN. The F-measure, precision, and recall measurements are shown in the table.

Table 5: The performance of the LSTM-DNN using different experimental settings

No.	Training	Testing	Recall	Precision	F-Measure
1	DD (40)	DD (40)	0.712	0.68	0.69
2	DD (16)	DD (16)	0.648	0.65	0.64
3	CCD (16)	CCD (16)	0.60	0.58	0.59
4	DD (16)	CCD (16)	0.625	0.61	0.61
5	CCD (16)	DD (16)	0.421	0.43	0.42
6	DD (16) + CCD (16)	CCD (16)	0.66	0.64	0.65
7	CCD (16) + DD (16)	DD (16)	0.57	0.55	0.56

5.8 Comparison of Proposed Method and State-of-the-Art Techniques

The proposed model is compared with state-of-the-art support vector machine (SVM) and decision tree algorithms. All seven experiments were repeated with SVM and decision tree. The recall, precision, and F-measure were computed for these techniques and compared with the proposed LSTM. The results show that the proposed model outperforms in every experiment. Tab. 6 shows the accuracy and F-measure for the proposed model, SVM, and decision tree algorithms.

Table 6: Comparison of performance of the LSTM-DNN with SVM and decision tree under different experimental settings

No.	Experiments	SVM	Decision tree	LSTM-DNN
	Training-testing	F-measure	F-measure	F-measure
1	DD (40)–DD (40)	0.42	0.37	**0.69**
2	DD (16)–DD (16)	0.39	0.38	**0.64**
3	CCD (16)–CD (16)	0.34	0.36	**0.59**
4	DD (16)–CD (16)	0.32	0.32	**0.61**
5	CCD (16)–DD (16)	0.29	0.31	**0.42**

6 Conclusions and Future Work

In this paper, experience modeling was employed using LSTM-DNN, which outperformed the analysis of time series physiological sensor data and predicted the emotional states of gamers during the game. A publicly available DEAP dataset was used for training the model, and testing data were collected for players while playing the game using the concept of transfer learning. Several experiments were conducted in order to validate the performance of the proposed LSTM-DNN over other machine learning techniques. The results were concluded in terms of performance accuracy using different sized datasets. The computational cost of processing huge amounts of data is reduced by selecting suitable sensor data that provide promising results. Although the resultant accuracy is slightly reduced, laborious feature engineering tasks are reduced, which burdens system speed and reduces performance over time. The proposed model not only solves the problem of predicting emotions but also provides a sustainable cue to increase performance. One such example is to merge the data of different domains for training. This can be done in a way that trains the model with data available for one domain. The trained model is tested for new data from another domain and trained again on the new domain data to further improve the performance of the model. In the future, this transfer learning approach can be used to improve the performance of predicting the experience of game players while playing games.

Acknowledgement: This study was supported by the BK21 FOUR project (AI-driven Convergence Software Education Research Program) funded by the Ministry of Education, School of Computer Science and Engineering, Kyungpook National University, Korea (4199990214394). This work was also supported by the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea Government (MSIT) under Grant 2017-0-00053 (A Technology Development of Artificial Intelligence Doctors for Cardiovascular Disease).

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. S. Farooq and K.-J. Kim, "Game player modeling," in *Encyclopedia of Computer Graphics and Games*, N. Lee (Ed.), Cham: Springer International Publishing, pp. 1–5, 2015.

[2] G. N. Yannakakis and J. Togelius, "Modeling players," in *Artificial Intelligence and Games*. Berlin, Germany: Springer, pp. 203–255, 2018.

[3] P. Lopes, A. Liapis and G. N. Yannakakis, "Modelling affect for horror soundscapes," *IEEE Transactions on Affective Computing*, vol. 10, no. 2, pp. 209–222, 2017.

[4] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[5] G. N. Yannakakis and J. Hallam, "Evolving opponents for interesting interactive computer games," *From Animals to Animats*, vol. 8, pp. 499–508, 2004.

[6] K. Majchrzak, J. Quadflieg and G. Rudolph, "Advanced dynamic scripting for fighting game AI," in *Int. Conf. on Entertainment Computing*, Cham, Switzerland, pp. 86–99, 2015.

[7] T. Mahlmann, J. Togelius and G. N. Yannakakis, "Modelling and evaluation of complex scenarios with the strategy game description language," in *2011 IEEE Conf. on Computational Intelligence and Games*, Seoul, South Korea, pp. 174–181, 2011.

[8] R. Cowie, C. Pelachaud and P. Petta, "Emotion-oriented systems," *Cognitive Technologies*, 1st ed., vol. 1, pp. 9–30, 2011.

[9] Delorme, T. Mullen, C. Kothe, Z. A. Acar, N. Bigdely-Shamlo *et al.,* "EEGLAB, SIFT, NFT, BCILAB, and ERICA: new tools for advanced EEG processing," *Computational Intelligence and Neuroscience*, vol. 11, no. 1, pp. 1–10, 2011.

[10] J. Valls-Vargas, S. Ontanón and J. Zhu, "Exploring player trace segmentation for dynamic play style prediction," in *Proc. of the Eleventh AAAI Conf. on Artificial Intelligence and Interactive Digital Entertainment*, California, USA, pp. 93–99, 2015.

[11] R. A. Calvo, I. Brown and S. Scheding, "Effect of experimental factors on the recognition of affective mental states through physiological measures," in *Australasian Joint Conf. on Artificial Intelligence*, Australia, pp. 62–70, 2009.

[12] F. Bisson, H. Larochelle and F. Kabanza, "Using a recursive neural network to learn an agent's decision model for plan recognition," in *IJCAI*, Argentina, pp. 918–924, 2015.

[13] S. J. Lee, Y.-E. Liu and Z. Popovic, "Learning individual behavior in an educational game: A data driven approach," in *Educational Data Mining*, London, United Kingdom, pp. 114–124, 2014.

[14] Zook and M. O. Riedl, "A temporal data-driven player model for dynamic difficulty adjustment," in *AIIDE*, California, USA, pp. 93–98, 2012.

[15] Pedersen, J. Togelius and G. N. Yannakakis, "Modeling player experience for content creation," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 2, no. 1, pp. 54–67, 2010.

[16] Bauckhage, A. Drachen and R. Sifa, "Clustering game behavior data," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 7, no. 3, pp. 266–278, 2015.

[17] G. Van Lankveld, P. Spronck and M. Rauterberg, "Difficulty Scaling through Incongruity," in *AIIDE*, California, USA, 2008.

[18] R. A. Bartle, "Designing virtual worlds: New riders," vol. 2, pp. 1–12, 2004.

[19] T. Mahlmann, A. Drachen, J. Togelius, A. Canossa and G. N. Yannakakis, "Predicting player behavior in tomb raider: Underworld," in *2010 IEEE Symp. on Computational Intelligence and Games*, Denmark, pp. 178–185, 2010.

[20] N. Shaker, G. N. Yannakakis and J. Togelius, "Towards automatic personalized content generation for platform games," in *AIIDE*, California, USA, 2010.

[21] E. Ha, J. P. Rowe, B. W. Mott and J. C. Lester, "Goal recognition with markov logic networks for player-adaptive games," in *AIIDE*, California, USA, 2011.

[22] Y.-E. Liu, T. Mandel, E. Butler, E. Andersen, E. O'Rourke *et al.,* "Predicting Player Moves in an Educational Game: A hybrid approach," in *EDM*, USA, pp. 106–113, 2013.

[23] K. Gold, "Training goal recognition online from low-level inputs in an action-adventure game," in *AIIDE*, California, USA, 2010.

[24] Melnik, S. Fleer, M. Schilling and H. Ritter, "Functional learning-modules for atari ball games," in *ICLR 2018 Workshop Track*, Vancouver, BC, Canada, pp. 1–4, 2018.

[25] N. Yee, "The demographics, motivations, and derived experiences of users of massively multi-user online graphical environments," *Presence: Teleoperators and Virtual Environments*, vol. 15, no. 3, pp. 309–329, 2006.

[26] B. Cowley, D. Charles, M. Black and R. Hickey, "Real-time rule-based classification of player types in computer games," *User Modeling and User-Adapted Interaction*, vol. 23, no. 5, pp. 489–526, 2013.

[27] K. Weiss, T. M. Khoshgoftaar and D. Wang, "A survey of transfer learning," *Journal of Big Data*, vol. 3, no. 1, pp. 9, 2016.

[28] G. Kuhlmann and P. Stone, "Graph-based domain mapping for transfer learning in general games," in *European Conf. on Machine Learning*, Greece, pp. 188–200, 2010.

[29] S. Sukhija, N. C. Krishnan and D. Kumar, "Supervised heterogeneous transfer learning using random forests," in *Proc. of the ACM India Joint Int. Conf. on Data Science and Management of Data*, Goa, India, pp. 157–166, 2018.

[30] D. Sobol, L. Wolf and Y. Taigman, "Visual analogies between atari games for studying transfer learning in RL," arXiv preprint, arXiv: 1807.11074, 2018.

[31] B. Spector and S. Belongie, "Sample-efficient reinforcement learning through transfer and architectural priors," arXiv preprint arXiv: 1801.02268, 2018.

[32] S. Koelstra, C. Muhl, M. Soleymani, J.-S. Lee, A. Yazdani *et al.,* "Deap: A database for emotion analysis; Using physiological signals," *IEEE Transactions on Affective Computing*, vol. 3, no. 1, pp. 18–31, 2012.

[33] L. Devillers, R. Cowie, J. Martin, E. Douglas-Cowie, S. Abrilian *et al.,* "Real life emotions in french and english TV video clips: An integrated annotation protocol combining continuous and discrete approaches," in *5th Int. Conf. on Language Resources and Evaluation*, Genoa, Italy, pp. 22, 2006.

[34] L. Guala, S. Leucci and E. Natale, "Bejeweled, candy crush and other match-three games are (NP-) hard," in *2014 IEEE Conf. on Computational Intelligence and Games*, Germany, pp. 1–8, 2018.

[35] C. Chen and L. Leung, "Are you addicted to candy crush saga? An exploratory study linking psychological factors to mobile social game addiction," *Telematics and Informatics*, vol. 33, no. 4, pp. 1155–1166, 2016.

[36] C. J. Larche, N. Musielak and M. J. Dixon, "The candy crush sweet tooth: How 'near-misses' in candy crush increase frustration, and the urge to continue gameplay," *Journal of Gambling Studies*, vol. 33, no. 2, pp. 599–615, 2017.

[37] S. Tripathi, S. Acharya, R. D. Sharma, S. Mittal and S. Bhattacharya, "Using deep and convolutional neural networks for accurate emotion classification on deap dataset," in *AAAI*, California, USA, pp. 4746–4752, 2017.

[38] L. Li, D. Wang, Y. Chen, Y. Shi, Z. Tang *et al.,* "Deep factorization for speech signal," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Canada, pp. 5094–5098, 2018.

[39] S. Salari, A. Ansarian and H. Atrianfar, "Robust emotion classification using neural network models," in *2018 6th Iranian Joint Congress on Fuzzy and Intelligent Systems*, Kerman, Iran, pp. 190–194, 2018.

[40] Y. Baveye, E. Dellandrea, C. Chamaret and L. Chen, "Liris-accede: A video database for affective content analysis," *IEEE Transactions on Affective Computing*, vol. 6, no. 1, pp. 43–55, 2015.

[41] B. U. Westner, S. S. Dalal, S. Hanslmayr and T. Staudigl, "Across-subjects classification of stimulus modality from human MEG high frequency activity," *PLoS Computational Biology*, vol. 14, no. 3, pp. e1005938, 2018.