Role of Human Emotions in Social Influence

$Pankaj\ Mishra$[1], $Rafik\ Hadfi$[2] and $Takayuki\ Ito$[3]
Department of Computer Science
Nagoya Institute of Technology
Gokiso, Nagoya, 466-8555
$pankaj.mishra@itolab.nitech.ac.jp$[1], $rafik@itolab.nitech.ac.jp$[2] and
$ito@itolab.nitech.ac.jp$[3]

Abstract

Diffusion of information can be found in any social network, either it be a small network of co-workers or large as social networking applications like Facebook, Twitter, LinkedIn, etc. In this paper, we propose a system to analyse the correlation amongst the group of people and build the influential relationship amongst them, which is based on the diffusion of emotions within this network. Actually we track the action and reaction in the form of facial emotions of all the participants in the network, and based on which we discover the influence among them. Later this influential relationship in the network is represented in weighted directed graph; where weight represents the extent of influence between each pair of nodes. Moreover, considering the robustness and scalability of the proposed algorithm with the varying size of the network, we have incorporated multiagent paradigm in our model. Further, our results were validated by analysing a scripted discussion done in our laboratory. Such knowledge of the influences among the participants of the network, could be used to find the influential individual; further can find many applications such in consensus, negotiation, viral marketing, etc.

Keywords: facial feature extraction; classification; correlation analysis

I Introduction

A social influence is the behavioral change of a person because of perceived relationship with other person or perceived action from the other person. In the recent years, many work were proposed to analyse the social correlation in the social network comprising of small group of people or large social networking applications like Twitter, Facebook, etc. Social correlation in the larger networks like Facebook, Twitter were analysed as discussed in the work [1, 2]; which tracks the online actions to build the correlation. Similarly, there are many work to analyse the social correlation based on the multimedia data of the workgroup, social gathering, etc., by tracking their body pose, interaction frequency, speaking and writing frequency etc., as discussed in [3, 4]. Briefly, most of the existing work to analyse the interaction in the small group, the social interactions focuses on the pose detection, namely head pose, shoulder pose, hand pose or their interaction interaction, etc. However, few attempts were taken to analyse the social influential correlation based on the facial emotions.

In this paper, we shall discuss a facial emotion based social correlation analysis among small group of people, comprising of co-workers in discussion, members in the business meeting, etc. In any such groups, people have the tendency to come together and form virtual groups. In these virtual groups, the actions of the influential person is propagated to others, mainly because of three factors; cofounding, homophile and social influence [5]. According to homophile and cofounding; two individuals are most likely to follow each other because of factors like common interest, friend, colleagues, etc. Such factors are mainly decided on their background information, such as their profession, knowledge of the topic of discussion, age, family ties,etc. Because, based on these parameters, we can find the existence of some kind of social ties two person; such as boss and employee relation, siblings, collogues, scholar of same field, etc. However, the analysis of influence is purely based on tracking the action and the corresponding reaction between any pair of persons; where, the action can be a pose, talk (spoken sentences), emotion, interactions, etc. In our work, we are focused on tracking the emotion and talking sequence of all the participants to analyse the influence amongst them. Also, facial emotions can be considered as one of the important action for human communication and understanding the social relationship between them, because when an influential person has a emotion people around him/her can read it and react according irrespective of the words spoken or any body pose. So in this work, we focus on human emotions to understand their correlation, also emotions are voluntary reaction on any action, so could depict the appropriate feeling of the person for an particular person in the network. In order to maintain the scalability [6] of the proposed algorithm, we have adopted multi-agent paradigm, where each participants in the network is treated as homogeneous autonomous agent.

In this paper, we have made two main contributions: (i) social influential correlation building algorithm based on emotion diffusion analysis in the network, (ii) profiling algorithm to label all participants based on the available data of each nodes. Apart from this, we the testing video data-set of scripted discussion could be used to build the similar emotion based systems.

A Architecture

The architecture of the system is depicted in the figure 1, which consist of two main modules, facial emotion analysis and social correlation analysis. A captured video data of the social

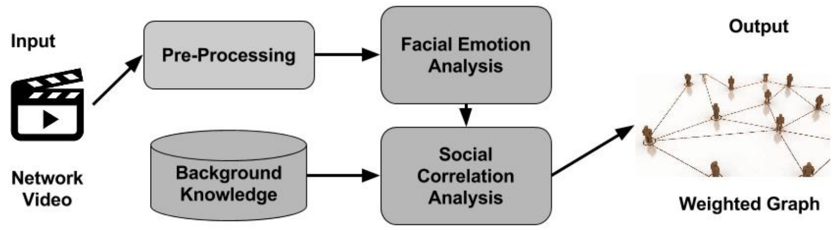


Figure 1: System Architecture

network is first pre-processed and later passed through these two modules. As previously mentioned, each individual is handled by homogeneous autonomous agents, these agent interact with the two modules; facial emotion analysis and social correlation analysis, independently. Finally a weighted graph representing the correlation among these agents, is obtained as the output of the system. Briefly, our whole work is divided into two main steps; facial emotion analysis and social correlation analysis, using the machine learning techniques and social dynamics concepts respectively. Later, implementation of the whole system is realized in multiagent paradigm.

The rest of the paper is organised as following. Section 2 explains the details about facial emotion analysis, which is divided into two main steps; feature extraction and emotion recognition by comparing different classifiers. Section 3 explains the social correlation analysis along with the all the node centrality algorithm and correlation building algorithms. Section 4 presents the experimental results that validate our proposed algorithm. Finally, we conclude and provide the future work.

II Facial Emotion Analysis

In the proposed work, the system expects a frontal video data of a group of participants discussing or in a business meeting. Wherein, change in one's emotion being induced on the other's emotions can be observed. This video data is first pre-processed; which includes : framing of the video and cropping the frontal image in all the frames. Locating and cropping is done using Voila Jones algorithm [7]. These, processed frontal frames for all the participants along with the talking sequence [1] of all the participants in each frame is recorded; where the actions include, emotion, talk sequence or both based on the metadata. After the pre-processing, we detect the facial emotion in all the frames in two stages, namely (i) Features extraction and (ii) Emotion classification, as discussed below.

A Facial features extraction

Feature extraction is done in two stages: tracking the facial feature points; namely eyes, eye brows, nose, lips, etc., and then extracting the shape and appearance feature vectors around

[1]Tracking when the participant is talking or not

these detected feature points. The shape feature vectors basically resembles co-ordinates (x and y) of all the facial features; whereas appearance features can be a Gabor descriptor or Local Binary Pattern (LBP) descriptor, etc as discussed in [8]. However, most of the past works like, [9, 10, 11], are based on the deformable model named Active appearance model (AAM). The AAM is a computer vision algorithm to track the facial feature points on human face, which provides a compact statistical representation of the shape and the appearance variation of it. Therefore, we adopted a variation of the AAM, that is a part based AAM [12] to track 68 facial-landmarks points in the frontal frames. Finally we extract four different types of the feature vectors: (i) shape vector resembling the x&y coordinates of all the facial features, called as similarity normalised shapes (s-pts), (ii) holistic appearance feature vector; called as similarity normalised appearance (s-app) and canonical appearance (c-app) [9], and (iii) A 32×32 patched vector called as scale-invariant feature transform (SIFT)[13] descriptors. Later, classifiers are trained with these feature vectors to classify the frames based on the carrying emotion.

B Emotion Classification

The four feature vectors associated with all the frontal frames are used to detect the emotion in the each frames. Basically, emotions are recognised by presence of one or more, particular Facial Action Units (FACS) [14]; further the combination of these FACS defines one of the 7 basic emotions; namely, happy, sad, disgust, anger, contempt, surprise and fear. The classifiers are trained with the feature vectors extracted from the frames, against the labels of FACS or combinations of FACS (emotion). These classifiers classify the frames based on the presence of FACS or emotions; therefore the accuracy of emotion recognition is mainly based on the accuracy of the emotion detection module. So, we compare different classifiers, and classifier that would give the best classification accuracy will be adopted. For that matter, we are comparing three classifiers, namely, Scalar Vector Machine (SVM) [15], Scalar Vector Machine with AdaBoost (SVM + Adb) [16] and the Deep Belief Network (DBN) [17]. Whereas, we will evaluate the classifiers based on cross validation and maximum area under the curve; discussed further in the experimental section. Briefly, facial emotion analysis module comprises of a trained AAM model and the best classifier amongst the three. Once all the frames are labeled with the carrying emotion, then these emotion labels are used for the social correlation analysis.

III Social correlation Analysis

In the previous section, we obtain the emotion labels associated with all the frames. However, for the experimental purpose we considered only the emotion of the person occurring per second, by summarising all the emotions occurred in all the frames per second.These emotion data of all the participants is used for social influential correlation analysis. The overall analysis of social correlation amongst the participants is based on the detected emotion per seconds (E), the tracked action per second (A) and the background knowledge (K). Wherein, K consist of whatever information about the participants in the network is available; such as age, gender, profession, seniority, family ties etc. Further, based on this background knowledge, we label all the individuals on the basis of their importance and affinity. Although, there are many different methods to calculate affinity (γ) or the node centrality in a network, as discussed in [18], in the considered domain, the mentioned methods do not suffice our purpose to label each participants as per the affinity between them. In such social network of group of people in social gathering, it is more appropriate to decide the affinity and importance of the node on various factors; such as, age, experience of work, knowledge about the work, position in the company, etc., that is the details present in *K*.

Algorithm 1 Nodal Affinity

1: **procedure** GAMMA
2: **Input:** L is number of labels
3: $X \rightarrow \{\{x_1\}, \{x_2\}, \ldots, \{x_n\}\}$ is set of n factor sets.
4: Each factor set is of size i; all i node
5: **Output:** γ for all the i node
6: **Start**
7: **for** all X $j \rightarrow 1$ to n **do**
8: $\{c^n\} \leftarrow$ **Cluster**($\{x_n\}$)
9: Sorting the labels
10: $\{l^n{}_i\}$ = **Sort**($\{c^n\}$)
11: $\gamma_i \rightarrow \frac{\sum_1^n l^n{}_i}{n}$
12: **End**

To rank the each individuals, we propose a simple clustering algorithm to find the affinity label (γ) for all the individuals; based on the background knowledge (K). The γ calculation algorithm is shown in algorithm 1; wherein clustering is based on *K-means* clustering algorithm. In the algorithm 1, *n* is the number of factors considered for affinity analysis. The *Cluster() function* is the *K-means* clustering algorithm. The *Sort() function* is the algorithm which sorts the affinity labels c^n for individual considered factors in ascending order of the data points in the cluster. Now, these sorted labels of all the factors are used to calculate the γ_i for all the individuals *i*, using the equation on line 11 of algorithm 1. At the end of the algorithm, we get γ for all the individuals in the network. Now, in any given social graph, emotion propagates among the nodes of the graph. This knowledge of emotion propagation helps to find the influential correlation amongst all the nodes. In our approach, this is done by analysis of change in one's emotion or action causing change in the emotions of others. The intuition behind the γ is, if one of the participant has some social ties with the other, then they tend to form the virtual groups and in this virtual group chances of induction is more as compared to other pairs without any social ties. However, mere γ does not contribute to the influential characteristics of a person.

Because, an influential character depends on many factors such as talking manner, expressions, way of conveying, credibility of the person, etc. So we use γ to guide the algorithm to locate the virtual groups and later analyse influential node in the virtual group first and then the rest of un related nodes. However, it can be said that, higher the γ , higher is the chances of the node to be an influential node in the particular virtual group. The whole process of social influence detection is divided into two basic steps (i) Emotion diffusion tracking and (ii) Nodal Social influence calculation, as discussed in the next subsections.

A Emotion Diffusion Tracking

In any given social graph, action and reaction interactions are propagated among the nodes of the graph, in our approach the interaction are in the form of facial emotion. This knowledge of emotion propagation helps to find the influential correlation amongst all the nodes in the network. Basically, our algorithm analyse the change in one's emotions because of change in the emotions of others. However, in order to consider the case where there is induction of emotion in absence of emotion of other nodes. Therefore we also considered the induction of emotion on a particular node by the action of the other node. We tracked the talking sequence of all the nodes; that is tracking the whether the node is talking or not talking, if the node is talking than node is called as active node or else inactive node. So, induction of emotion is analysed in two different scenarios: induction by action $(a \rightarrow e)$ and induction emotion $(e \rightarrow e)$. In $a \rightarrow e$ scenario, emotions of active node induces emotion on the other nodes; whereas in $e \rightarrow e$ emotions of a node (irrespective of its being active of inactive) induces emotion on other nodes. The whole idea in considering the two scenarios was to refrain from considering wrong orientation of influence between the pair of nodes. Because, may be the change in emotion of a particular node is because of actions of particular node and building the correlation merely based on the emotions may lead to wrong interpretation. Now, the diffusion parameter (ΔE) for both the scenario is calculated using the equation 1.

$$\Delta E = \frac{\tau_e \times \omega_e}{T} \tag{1}$$

In the equation 1, τ_e is the time interval of a particular emotion *e*, and ω_e is a participant's emotion coefficient for a particular emotion *e*. The T is the total frame considered for emotion diffusion calculation ΔE. Whereas, ω_e is calculated using the equation 2, which is pre-calculated for all the participants for every emotion. t_e is the total time instance, for which participant has an emotion *e* and N_e is the total number of instances, when all the participants has emotion *e*.

$$\omega^e = \frac{\gamma \times t^e}{N^e} \tag{2}$$

Other than this, we define the orientation of emotion diffusion by δ, which can have value

1 or -1, based on the orientation of the induced emotion. The δ for all the considered emotion is listed in the table 3, which is based on the discussion in [19]. The emotion diffusion in both the scenarios is calculated using the algorithms 2 and 3. Let us first discuss the algorithm 2 for $a \to e$ scenario, which accepts the list of inactive (participant with action: emotion or no emotion) agents for every active (agent with action: speech or both) agents, and gives $\Delta E'$ and δ' for all combination of passive and active participants as output. In the equation on line 8, we calculate the $\Delta E'^e$ values for each emotion e carried by the participant p for time τ^e, during the interval T_a, for which the participant *a* was active. Then, the maximum value in set $\Delta E'^e$ is chosen as the final $\Delta E'$. Thus, from the above we deduce that emotion e was induced on agent *p* by agent *a*. Later final δ' is calculated by multiplying the δ of *a* and *e*, which gives the direction of the emotion induction. Similarly, in algorithm 3, emotion diffusion $\Delta E''$ and δ'' are calculated for all the pairs of nodes, irrespective of the active participant, where value of *T* is 1. At the end of this step, we get four values, $\Delta E'$ and $\Delta E''$ denote the emotion diffusion parameters and δ' and δ'' denote the direction of emotion diffusion values for all the pairs of participants. Then, we build the influential correlation amongst the participants, as discussed in the next subsection.

Algorithm 2 Emotion Propagation $(a \to e)$

1: **procedure** EMOTION DIFFUSION
2: **Input:** a Active participant
3: P List of passive participant
4: **Output:** $\Delta E'$ and δ' for all passive participant
5: (where δ is -1 or 1)
6: **for** all p enumerate(P) **do**
7: **for** all e Emotion E **do**
8: $\{\Delta E'^e{}_{ap}\} \to \frac{\tau^e{}_{ap} \times \omega^e{}_p}{T_a}$
9: (Maximum of $\Delta E'^e$ is final $\Delta E'$)
10: (δ^e is emotion having maximum for $\Delta E'^e$)
11: $\Delta E'_{ap} \to max(\{\Delta E'^e{}_{ap}\})$
12: $\delta'_{ap} \to \delta_a \times \delta^e{}_p$
13: **End**

B Nodal Social Influence

From the above algorithms 2 and 3, we get the emotion diffusion $\Delta E'$,$\Delta E''$, $\delta'$and δ'' for both the scenarios. From this we calculate the influence of each agent on the other agents, based on the algorithm 4. The objective is to find the social influence amongst all the agents in the network, in terms of weight of the weighted graph. The input to this algorithm is list of $\Delta E'$, $\Delta E''$, δ' and δ'' for all the pairs of agents.

Based on these values, the weight W_{ij} between agent *i* and *j* is calculated using the equation on line 7. Where, summation of $\Delta E'$ and $\Delta E''$ for *i* and *j* is divided by summation of $\Delta E'$ for

Algorithm 3 Emotion Propagation $(e \rightarrow e)$

procedure EMOTION DIFFUSION
  Input: A List of all participant
  B List of all participant
  Output: $\Delta E''$ and δ'' for all pair of participants
  (where δ is -1 or 1)
  for all a enumerate(A) **do**
    for all b enumerate(B) **do**
      for all e Emotion E **do**
        $\{\Delta E''^{e}{}_{ab}\} \rightarrow \frac{\tau^{e}{}_{ab} \times \omega^{e}{}_{b}}{T}$
        (where T is 1)
    (Maximum of $\Delta E''^{e}$ is final $\Delta E''$)
    (δ^{e} is emotion having maximum for $\Delta E''^{e}$)
    $\Delta E''_{ab} \rightarrow max(\{\Delta E''^{e}{}_{ab}\})$
    $\delta''_{ab} \rightarrow \delta_{a} \times \delta^{e}{}_{b}$
  End

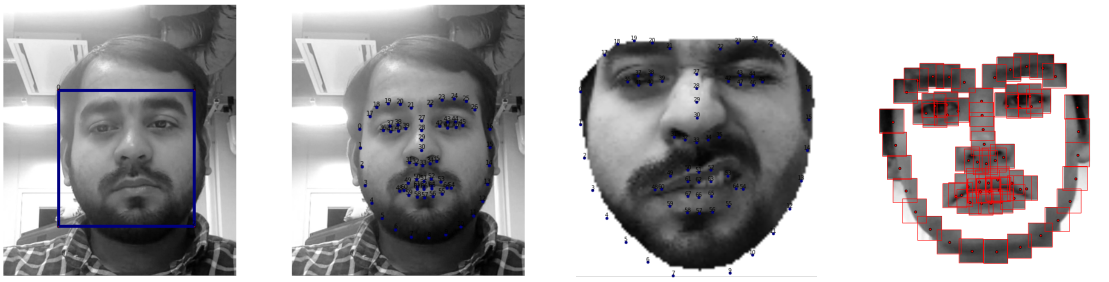
all the neighbouring nodes of i. Finally, the orientation of the influence is associated with the W_{ij}, by multiplying δ_{ij}. Thus, as an output we get a weighted graph, where weight represents the influence and sign '-' and '+' represents the orientation of the influence being induced.

Algorithm 4 Influence Calculation

procedure INFLUENCE
  Input: $\Delta E'$ and $\Delta E''$ for each pair of nodes
    δ' and δ'' for each pair of nodes
  Output: W weight for each pair of nodes
  for each pair of node i and j **do**
    $\delta_{ij} \rightarrow \delta'_{ij} \times \delta''_{ij}$
    $W_{ij}= (\frac{\Delta E'_{ij}+\Delta E''_{ij}}{\sum_{k \in N(i)} \Delta E'_{ik}}) \times \delta_{ij}$
    (N(i) neighbours of node i)
  End

IV Experimental Results and Discussion

This section reports experimental results and evaluating the obtained results by comparing it with the ground truth of the testing data. Our system analyses the social influence in the network, based on change in facial expression amongst the participants of the network. Our system is implemented in a multiagent paradigm, where all the participants are represented by homogeneous agents; these autonomous agents interact with the different modules and finally correlation is build amongst them. The facial emotion analysis module is build up of AAM based facial feature extraction and a classifier. The AAM is implemented in *Menpo* [20] python libraries, and trained with two publicly available databases; FERA database [21] and LFPW database [22]. Now, different feature vectors are extracted by tracking the feature landmarks using the trained AAM is used for training the classifier. The tracked frontal frame using the Viola Jones algorithm is depicted in figure 2a, and the figure 2b depicts the tracked features points (landmarks) and the x&y co-ordinates are the shape features a.k.a. similarity normalised shapes. Whereas, figures 2c and 2d depicts the appearance features (s-aap) and patched based features respectively; the *c-aap* appearance features are similar to *s-aap* without any facial feature movement.



(a) Frontal Face

(b) Similarity normalised shapes (s-pts)

(c) Similarity normalised appearance (s-aap)

(d) Patched SIFT descriptor

Figure 2: Feature Extraction

Later, these feature vectors are feed in to the trained classifier, to classify the frames on the basis of carrying set of FACS. As mentioned before, we compared 3 classifier‘s, and adopt the comparatively accurate classifier. The classifier *SVM* and *SVM+Adb* is implemented with python API; whereas, DBN implementation is based on the DBN for classifying the MINST handwritten digital database [17], because problem of classifying handwritten images is similar to the emotion classification.

In our experiment, we have only considered 5 emotions; namely, neutral, happy , sad, surprise and angry. The training and testing of the classifiers are done using the publicly available CK+ database [11]. The table 1 enlist the scores of different classifier for all the different emotions; we adopted *SVM+Adb* for emotion classification as it can be seen that the score of *SVM+Adb* classifier is more for most of the emotions. The table 3 enlist the combination of FACS for each considered emotions along with their orientations δ.

In order to test our proposed algorithm, we need a input frontal video data of small network to detect the emotions and later build the correlation. However, such dataset is not available, therefore a small experiment was conducted in our laboratory, we record a discussion of five participants on a video conference; let us say them as participants A, B, C, D and E. Additionally, the video should be able to showcase the influence caused by the emotion and an action. Therefore, we structured the script, such that each participants does the action (talks) for approximately 90 seconds one by one, yielding a total of 450 seconds video.

Further, the script also shows that the person A, B and C belongs the same organisation and D and E belongs to other organisation. Also, A and D are seniors amongst their group and

B has the higher knowledge on topic of discussion. Based on these assumptions of seniority, knowledge about the topic, etc, and the algorithm 1 we get 3, 5, 4, 5 and 2 as γ_A, γ_B, γ_C, γ_D and γ_E respectively.

Once video is pre-processed, each individual in the network is associated with an autonomous and homogenous agent. These agents first interact with the emotion analysis and the four types of feature vectors are extracted using the AAM tracker.

Later, these vectors are used to recognise the emotions carried by each agent (frame by frame), by classifying it with *SVM+Adb* classifier. Basically, we used binary classifiers; 5 classifiers were used, one for each emotion. Whereas, the approximate dimensions of *s-pts, s-aap, c-aap* and patched based features are 136 (x&y coordinates of 68 landmark points), 56000, 56000 and 132000 (approximately) respectively.The table 2 represents the emotions detected per second for all the agents at the end of classification of all the 450 frames. Now, the social correlation analysis for both the scenarios is done using the algorithm 2 and 3. The table 4 and 5 enlist the obtained ΔE^1 and ΔE^2 for both the scenarios. Further based on the algorithm 4 and diffusion parameters, we calculate the influence between the pair of nodes, in terms of weights of weighted graph; listed in the table 6. The evaluation and validation of the results is done on the basis of the ground truth of the input data, as discussed in the next subsection.

FACS	SVM	Adb+SVM	DBN
Neutral	80	93	75
Happy	81	92	69
Surprise	73	83	75
Sad	79	85	70
Angry	70	82	65

Table 1: Emotion detection on CK+ dataset

Agent	Neutral	Happy	Surprise	Angry	Sad
A	100	20	150	35	145
B	45	190	75	100	40
C	30	140	80	125	75
D	27	190	50	153	30
E	58	140	80	112	60

Table 2: Detected Emotion per seconds

A Discussion

The input data to our system is the scripted group discussion of 5 persons (nodes); wherein the roles and responses of each node are known. This information gives idea about the social ties between the nodes in the network. Further in to build the ground truth of the fed data, we

Label	Emotion	δ	FACS
1	Neutral	+1	null
2	Happy	+1	6+12
3	Surprise	+1	1+2+4+25
4	Sad	-1	1+4+15
5	Angry	-1	4+7+23+25

Table 3: Emotion Orientation (δ) and FACS

Agent	A	B	C	D	E
A	0	1.8	0.9	0.4	0.8
B	2.5	0	2.2	1.5	1.2
C	0.8	1.5	0	0.7	0.4
D	0.6	1.2	0.4	0	3.2
E	0.8	0.8	0.2	1.8	0

Table 4: $\Delta E'$

Agent	A	B	C	D	E
A	0	2.5	1.2	0.8	0.5
B	2.8	0	2.5	1.2	0.8
C	1.2	1.5	0	0.2	1.2
D	0.8	1	0.7	0	2.8
E	0.4	0.6	0.2	1.8	0

Table 5: ΔE"

Agent	A	B	C	D	E
A	0	-0.81	0.75	0.3	0.27
B	-2.4	0	3.13	-0.93	0.45
C	0.512	0.53	0	0.24	0.3
D	0.34	-0.372	0.33	0	2.5
E	0.3	0.22	0.114	1.38	0

Table 6: Final Weights

analyse the change of emotions of each node with respect to other for all time instances. In order to do that, we plot the change of emotion with time for all nodes in both the scenarios: $a \rightarrow e$ and $e \rightarrow e$. The emotion propagation curves 3 and 4 represents scenario $a \rightarrow e$ and $e \rightarrow e$ respectively; where 3 represents the emotion change curves for all the 5 nodes with respect to one node is active and 4 represents the emotion change curve for throughout the discussion irrespective of any node is active. The *y-axis* denotes the 5 emotion labels normalized along the axis and *x-axis* denotes time in second. If we analyse curves for both the scenario point by point and emotion labels listed in table 2, we could build the ground truth of existing correlation in the considered network. Firstly, from the emotion propagation curve 3 in $a \rightarrow e$ scenario, It can be said that emotion diffusion of B induces an positive emotion change on C ofently. Also, B induces negative emotion change on node A, i.e., when A is happy then most likely the node B is sad. Similarly, it can be said that emotions of E is more often influenced by emotions of D. Whereas from the emotion propagation curve in the figure 4 for $e \rightarrow e$ scenario, It can be said that, B induces emotion on C and A; also induction of emotions in reverse sense is not reciprocated from the curves. Such correlations can also be observed between agent D and agent E. Now next step is, based on these key observed characteristic in the graph, we build the ground truth. Further, in order to validate the predicted influential correlation among the individuals, we simply compare the ground truth conclusion of correlation with the obtained correlation.

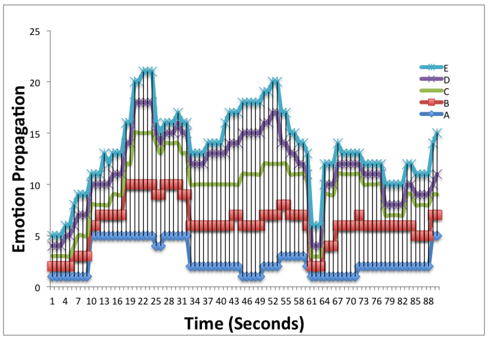
Lets analyse the predicted weights (influence) listed in the table 6. From the weights, it can be said that node B induces node A in the opposite orientation with weight -2.4, whereas agent B induces C to a greater extent; that is 3.13. Similar correlation is observed between node D and node E. Node D induces node E with weight 2.5; although, vice versa correlation is not observed in both the cases. Other than this, node B induces node D in negative sense. It can also be said that, if the value of weight is less, that means it has lower induction; then that edge or correlation can be ignored or removed form the network. Therefore, we set a threshold value for weights to be considered, the weights below that threshold is avoided in the final influence graph. As depicted in the figure 5, influence on nodes E $\rightarrow$ C, C $\rightarrow$ D, A $\rightarrow$ D and E $\rightarrow$ B are ignored because the influence is less then set threshold value, that is value less than 0.3. Thus, it will reduce the complexity of other searching algorithms (if any) in the network. Finally, by analysing the obtained results, it can be said that the emotion propagation in the scripted discussion is very well represented in terms of weights of the weighted graph. Therefore, it can also be said that our algorithm proves to be efficient to measure the influence in the network.

Such system can find many application such as negotiation or consensus building among the group of people, also finding the influential person can help the organization to target the appropriate person, who can persuade large group with its influencing capability.

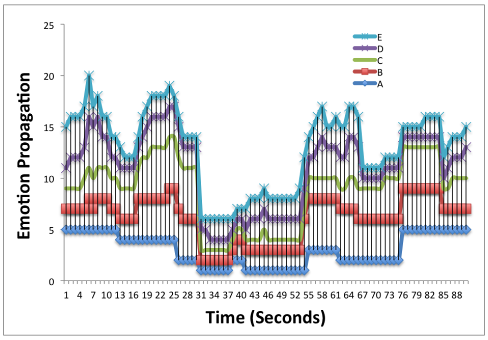
V Conclusion

We introduced an algorithm to analyse the influential correlation in the small social networks formed by group of people in discussion or business meetings in a multiagent paradigm. Wherein, we were more interested in understanding the role of human facial emotion on social influential correlation and how effectively a person can induce his/her emotion on other person. So, If any particular participant in the network brings a high degree of emotion change on the other participant/s in the network, then it can be said that the participant is an influential node in the network.

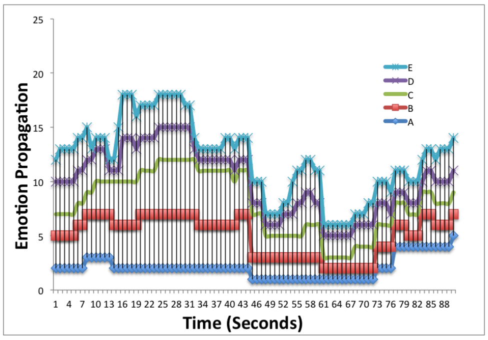
In our system, humans emotions were considered for analysing the influential analysis; whereas emotion recognition was based on the AAM feature tracking and the *SVM+Adb* clas-



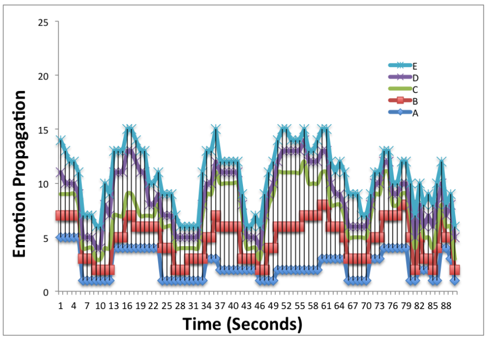
(a) Active Node: A



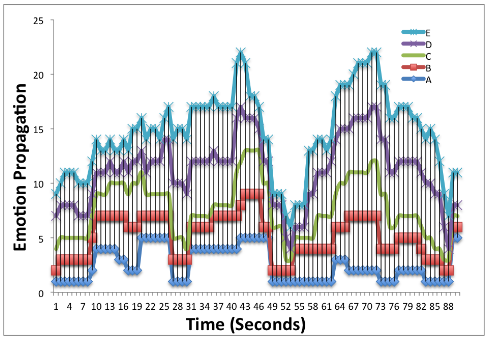
(b) Active Node: B



(c) Active Node: C



(d) Active Node: D



(e) Active Node: E

Figure 3: Emotion Propagation with Active Nodes

sifier for emotion classification. Also, we compared three classifiers; namely, *SVM+Adb*, *SVM* and *DBN* and choose the comparatively good classifier for our system. Our algorithm for influential correlation building is based on emotion diffusion parameter and node centrality calculation as discussed in the above sections. From these algorithms, we should be able to determine one influential person in the whole group or multiple influencer for each virtual groups in the network. Finally, we evaluated and validated our system based on the ground truth of the video data of a scripted discussion, fed in to the system. Concluding that the final output of our system was able to reflect the emotion propagation in the scripted discussion. As a future work, it would be interesting to extend our method to larger real life networks. Moreover, we are thinking of incorporating emotion from speech data along with facial emotion, so that the resultant emotion can depict humans emotion more closely and be used for consensus building. We also plan on investigating how social association can be build on any anonymous network by different social network mining methods.

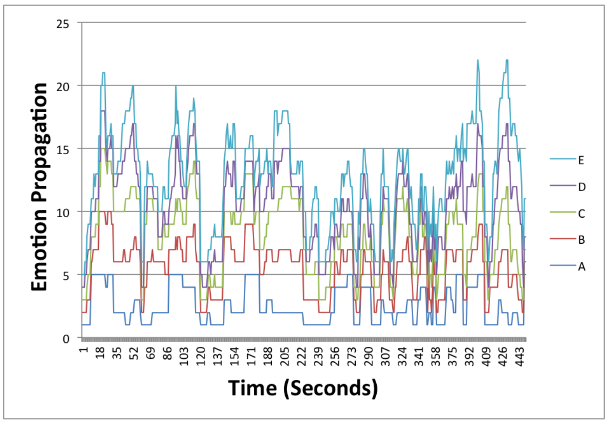


Figure 4: Overall Emotion Propagation

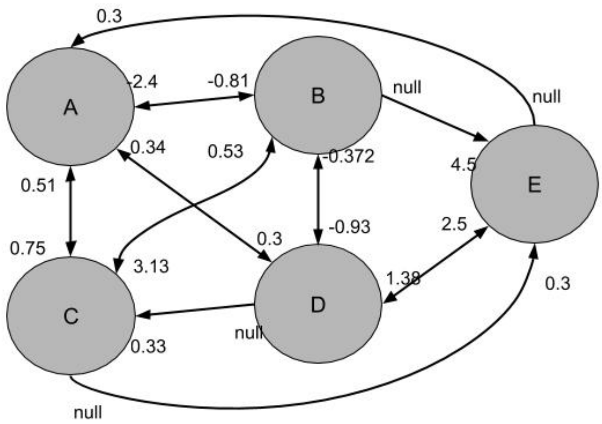


Figure 5: Influence Graph

Acknowledgments

This work has been partially supported by the project Large-scale Consensus Support System based on Agents Technology in the research area Intelligent Information Processing Systems Creating Co-Experience Knowledge and Wisdom with Human- Machine Harmonious Collaboration of JST CREST projects.

References

[1] J. Tang, J. Sun, C. Wang, and Z. Yang, "Social influence analysis in large-scale networks," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2009, pp. 807–816.

[2] A. Anagnostopoulos, R. Kumar, and M. Mahdian, "Influence and correlation in social networks," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2008, pp. 7–15.

[3] C.-W. Chen, R. C. Ugarte, C. Wu, and H. Aghajan, "Discovering social interactions in real work environments," in *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*. IEEE, 2011, pp. 933–938.

[4] D. Zhang, "Probabilistic graphical models for human interaction analysis," IDIAP, Tech. Rep., 2006.

[5] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annual review of sociology*, pp. 415–444, 2001.

[6] P. Stone and M. Veloso, "Multiagent systems: A survey from a machine learning perspective," *Autonomous Robots*, vol. 8, no. 3, pp. 345–383, 2000.

[7] O. H. Jensen, "Implementing the viola-jones face detection algorithm," Ph.D. dissertation, Technical University of Denmark, DTU, DK-2800 Kgs. Lyngby, Denmark, 2008.

[8] S. Happy and A. Routray, "Robust facial expression classification using shape and appearance features," in *Advances in Pattern Recognition (ICAPR), 2015 Eighth International Conference on*. IEEE, 2015, pp. 1–5.

[9] A. B. Ashraf, S. Lucey, J. F. Cohn, T. Chen, Z. Ambadar, K. M. Prkachin, and P. E. Solomon, "The painful face–pain expression recognition using active appearance models," *Image and vision computing*, vol. 27, no. 12, pp. 1788–1796, 2009.

[10] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 6, pp. 681–685, 2001.

[11] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*. IEEE, 2010, pp. 94–101.

[12] W.-S. Chu, F. Torre, and J. Cohn, "Selective transfer machine for personalized facial action unit detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3515–3522.

[13] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2. Ieee, 1999, pp. 1150–1157.

[14] P. Ekman and E. L. Rosenberg, *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, 1997.

[15] C.-W. Hsu, C.-C. Chang, C.-J. Lin *et al.*, "A practical guide to support vector classification," 2003.

[16] X. Li, L. Wang, and E. Sung, "Adaboost with svm-based component classifiers," *Engineering Applications of Artificial Intelligence*, vol. 21, no. 5, pp. 785–795, 2008.

[17] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[18] J. Sun and J. Tang, "A survey of models and algorithms for social influence analysis," in *Social network data analytics*. Springer, 2011, pp. 177–214.

[19] K. Ghamen and A. Caplier, "Positive and negative expressions classification using the belief theory," *International Journal of Tomography & Statistics*, vol. 17, no. S11, pp. 72–87, 2011.

[20] J. Alabort-i Medina, E. Antonakos, J. Booth, P. Snape, and S. Zafeiriou, "Menpo: A comprehensive platform for parametric image alignment and visual deformable models," in *Proceedings of the ACM International Conference on Multimedia*, ser. MM '14. New York, NY, USA: ACM, 2014, pp. 679–682. [Online]. Available: http://doi.acm.org/10.1145/2647868.2654890

[21] M. F. Valstar, B. Jiang, M. Mehu, M. Pantic, and K. Scherer, "The first facial expression recognition and analysis challenge," in *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*. IEEE, 2011, pp. 921–926.

[22] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: The first facial landmark localization challenge," in *Computer Vision Workshops (ICCVW), 2013 IEEE International Conference on*. IEEE, 2013, pp. 397–403.

[23] T. Simon, M. H. Nguyen, F. De La Torre, and J. F. Cohn, "Action unit detection with segment-based svms," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 2737–2744.

[24] T. Finin, Y. Labrou, and J. Mayfield, "Kqml as an agent communication language," *Software Agents. MIT Press, Cambridge*, vol. 239, 1995.

[25] A. Ruiz, J. Van de Weijer, and X. Binefa, "From emotions to action units with hidden and semi-hidden-task learning," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3703–3711.

[26] F. Bellifemine, F. Bergenti, G. Caire, and A. Poggi, "Jadea java agent development framework," in *Multi-Agent Programming*. Springer, 2005, pp. 125–147.

[27] Y. Zhu, F. De la Torre, J. F. Cohn, and Y.-J. Zhang, "Dynamic cascades with bidirectional bootstrapping for action unit detection in spontaneous facial behavior," *Affective Computing, IEEE Transactions on*, vol. 2, no. 2, pp. 79–91, 2011.