Facial Emotional Expressions of Life-like Character Based on Text Classifier and Fuzzy Logic

Surya Sumpeno, *Member, IAENG,* Mochamad Hariadi and Mauridhi Hery Purnomo

***Abstract*—A system consists of a text classifier and Fuzzy Inference System (FIS) to build a life-like virtual character capable of expressing emotion from a text input is proposed. The system classifies emotional content of sentence(s) from text input and expresses corresponding emotion by a facial expression. Text input is classified using the text classifier, while facial expression of the life-like character are controlled by FIS utilizing results from the text classifier. A number of text classifier methods are employed and their performances are evaluated using Leave-One-Out Cross Validation.**

In real world application such as animation movie, the life-like virtual character of proposed system needs to be animated. As a demonstration, examples of facial expressions with corresponding text input as results from the implementation of our system are shown. The system is able to show facial expressions with admixture blending emotions. This paper also describes animation characteristics of the system using neutral expression as center of facial expression transition from one emotion to another. Emotion transition can be viewed as gradual decrease or increase of emotion intensity from one emotion toward other emotion. Experimental results show that animation of life-like character expressing emotion transition can be generated automatically using proposed system.

***Index Terms*—Affective Computing, Text Classification of Emotion in Text, Facial Expression of Life-like Character**

I. INTRODUCTION

THE interest in computational models of emotion and emotional expressions has been steadily growing in the agent research community. Several psychologists have acknowledged the role of emotions in intelligence [1]. Minsky stated that, "the question is not whether intelligent machines can have any emotions, but whether machines can be intelligent without any emotions" [2].

The general vision here is that if a machine could recognize a user's emotion, the interaction between man and machine would become more natural, more enjoyable and comfortable experience for humans [3]. The machine, i.e. computer, could offer help or assistance to a confused user, try to cheer up a frustrated user, or simply empathize with the user's situation.

Life-like character convincingly implements the "computer as social actor" metaphors as its modalities include affective speech, facial emotional expressions, hand gestures, head movements and body posture. It is designed to establish socio-emotional relationships with human user. Since life-like character is endowed with some tools to express emotions, it is genuinely able to display (artificial) empathy to the human user. It is believed that life-like character technology may significantly improve human computer interaction.

Application of this research can be found in the next generation of intelligent robotics, virtual human, NPC (Non Player Character) in game, psychology, blogs, product reviews, to support development of emotion-ware applications such as emotion-ware Text-to-Speech (TTS) engines for emotional reading of text and virtual chat friend.

In this paper, the machine (a computer) is a life-like character which capable of understanding text input as part of a fully functional Embodied Conversational Agent (ECA). Other than its conversational skills, the non-verbal behavior and the appearance of ECA becomes more important and more realistic in the future. ECA offers great promise to more natural interaction in social settings like tutoring or gaming.

The first step of this application is Human Emotion Recognition (HER). In HER, the data collected to recognize human emotion is often similar to the signals that human use to perceive emotions of others. Hence, HER is naturally multimodal. It includes textual, visual (graphical) and acoustic (sound) features. The study of text-based emotion mostly done due to text form is relatively simple compared to other latter forms. HER from text can be considered as a classification task of a given text according to predefined emotional classes.

Once the emotions carried by text-based sentence have been classified, and intensities of emotions in text have been quantified, by a text classifier, the embodied agent i.e. life-like character, will respond non verbally using facial expressions. This facial expression is controlled by mechanisms based on Fuzzy Inference System (FIS) which receive probability inputs from the output of text classifier.

This paper is organized as follows: section II presents related works regarding text categorization task, relations between emotions and facial expressions, and computational models on emotional facial expressions. Section III describes the design of our proposed system, while section IV discusses experiments which subdivided into text classification and facial expression of life-like character, then followed by conclusion and future work in section V.

II. FROM TEXT TO EMOTIONAL FACIAL EXPRESSIONS

Text is not only conveying information, but also able to trigger emotional response in the reader (listener) or writer (speaker). For example, if someone reads headline of news article "Plane carrying 51 crashes in Venezuela; 36 survive"[1],

Surya Sumpeno, Mochamad Hariadi and Mauridhi Hery Purnomo are with the Electrical Engineering Department, Institut Teknologi Sepuluh Nopember (ITS), Surabaya, Indonesia, 60111 e-mail:(see http://blog.its.ac.id/surya).

[1]http://technews.tmcnet.com/topics/associated-press/articles/102174-plane-carrying-51-crashes-venezuela-36-survive.htm published at: September 16, 2010

(Advance online publication: 25 May 2011)

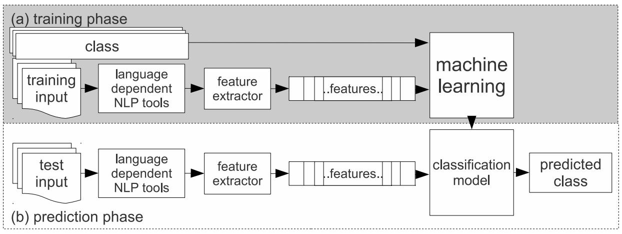

Fig. 1. Supervised Text Classification. Input is first preprocessed using language-dependent tools, such as tokenizing, stop words removal and (optionally) lemmatization. (a) During training, a feature extractor is used to convert each input value to a feature set. Pairs of feature sets and classes are fed into the machine learning algorithm to generate a classification model. (b) During prediction, the same feature extractor is used to convert unseen inputs to feature sets. These feature sets are then fed into the classification model, which generates predicted class. During cross-validation, these predicted classes will be matched against ground-truth classes.

he/she will feel sad and perhaps fear. Possible response could be emotional expression which is observable verbal and non-verbal behaviour. To create truly life-like socially believable character, non-verbal communication is a vital element to bring into it. According to [4] this emotional response is an essential component of any believable life-like character, character that provides the illusion of life. The non-verbal behavior, i.e. facial expression, is the main focus of this paper.

In order to be able to produce correct emotional response after receiving text input, firstly, life-like character must identify emotional content of an input sentence. This process can be considered as an automated categorization task of text.

A. Text Categorization

Text categorization tasks can be divided into two sorts: *unsupervised* text categorization (*text clustering*), where the categorization must be done entirely without reference to external information, and *supervised text classification*, where some external mechanism (such as training by human) provides information on the correct classification for documents. Text clustering aims to seek natural groupings, and thus presents an overview of the natural classes in a collection of documents. While automatic text classification is a process where the number of classes are known a-priori and documents are then assigned to these classes [5].

Most popular approach to automatic text classification is based on supervised machine learning technique: an inductive process automatically builds a classifier by learning, from a set of pre-classified (labelled) documents, provided by human user. The advantages of machine learning approach over the knowledge engineering approach (consisting in the manual definition of a classifier by domain experts) are a very good effectiveness, considerable savings in terms of expert manpower, and straightforward portability to different domains [6].

Automatic text classification has been used in many applications such as e-mail filtering [7], topic identifications [8], automatic meta-data organization, text filtering and documents' organization for databases and web pages [9], also emotion classification of text [10], [11].

In our proposed system, first data is preprocessed using dependent-language techniques such as tokenizing, stop words removal and optionally lemmatization, before fed into machine learning algorithm. The structure of the text classifier is depicted in Fig. 1.

Many available techniques using supervised learning algorithms for text classification have showed reasonable performance. These techniques include Simple Naïve Bayes (SNB) [12] with its variants such as Multinomial Naïve Bayes (MNB) [13] and Complement Naïve Bayes (CNB) [14], k-Nearest Neighbour (k-NN) [15], Vector Space Model (VSM) [10], Support Vector Machine (SVM) [16], boosting [17], rule learning algorithms [18] and Maximum Entropy [19].

Following sub-sections will describe further about Naïve Bayes approach and Vector Space Model, especially Term Frequency (*TF*) and Inverse Document Frequency (IDF).

1) Naive Bayes: Let $C = \{0, 1, \ldots, c_k, \ldots, |C|\}$ be the set of possible classes for a set of document $D = \{d_1, d_2, \ldots, d_{|D|}\}$, and $W = \{w_1, w_2, \ldots, w_m\}$ be a dictionary of ordered unique words (terms). It is assumed that there are m unique words in vocabulary of words W. A document d is represented by the vector $x = (x_1, x_2, \ldots, x_m)$.

A widely used NB for text classification is provided by a simple theorem of probability known as *Bayes' rule* which in its simplified form is:

$$P(c_k|x) = P(c_k) \times \frac{P(x|c_k)}{P(x)}, \tag{1}$$

where

$$P(x|c_k) = \prod_{j=1}^{m} P(x_j|ck). \tag{2}$$

It is assumed that all possible documents fall into exactly one of $|C|$ classes. In practice, multi-class NB classifies a document d to a single class c whose $\operatorname{argmax}_c \left[P(c_k) \times \prod_{j=1}^{m} P(x_j|c_k)\right]$ omitting the denominator $P(x)$, or if using logarithmic, $\operatorname{argmax}_c \left[\log(P(c_k)) + \sum_{j=1}^{m} \log(P(x_j|c_k))\right]$.

Utilising Bag-of-Word (BoW) approach, feature value x of document is the number of word w (term) occurrence in document, which one unique word (term) represents one feature. Word occurrence is referred as *tf* (term frequency). This is our definition of SNB.

In the MNB model [13], a document d_i is an ordered sequence of word events w_t, drawn from the vocabulary W.

$$P(d_i|c_k;\theta) = P(|d_i|)|d_i|! \prod_{t=1}^{|W|} \frac{P(w_t|c_k;\theta)^{N_{it}}}{N_{it}!}, \tag{3}$$

where N_{it} is the count of the number of times word (*tf*) w_t occurs in document d_i and probability of word w_t in class c_j is estimated by:

$$\hat{\theta}_{w_t|c_k} = P(w_t|c_k;\hat{\theta}_k) = \frac{1 + \sum_{i=1}^{|D|} N_{it} P(c_k|d_i)}{|V| + \sum_{s=1}^{|V|} \sum_{i=1}^{|D|} N_{is} P(c_k|d_i)}. \tag{4}$$

Apply Bayes' rule for text classification:

$$P(c_k|d_i;\hat{\theta}) = \frac{P(c_k|\hat{\theta}) P(d_i|c_k;\hat{\theta}_j)}{P(d_i|\hat{\theta})}, \tag{5}$$

then select the class whose the largest value of numerator as assigned class for the document d_i using argmax_c.

[14] describes systemic errors occurring in any NB text classifiers and proposes a better text model, which essentially

consists of three transforms: *tf* transform with smoothing, Document Frequency (DF) transform and transformation based on documents' length.

Interesting characteristic of Complement NB (CNB) as suggested by its name is "*complement* class" version of NB. In weighting estimation, "regular" MNB uses training data from a single class c_k to decide whether a document should be classified in c_k class. In contrast, CNB estimates parameters using data from all classes, *except* c_k. The results are probability values which show how *less* probable a document should *not* be classified to class c_k. That is why, in classifying, instead of argmax_c CNB uses argmin_c.

2) Vector Space Model: In VSM or term vector model, each document is represented as a vector, and each dimension corresponds to a separate term.

An arbitrary document vector d_i is defined as $d_i = (\mathbf{w}_{1,i}, \mathbf{w}_{2,i}, \ldots, \mathbf{w}_{p,i})$ where $\mathbf{w}_{k,i}$ represents the weight of k^{th} term (word w) in document i. One of the most popular technique for weighting is *TF·IDF* (Term Frequency-Inverse Document Frequency).

TF represents how important a term (word) is to a document in a set of documents or corpus, on the other hand, IDF is to discount weight of term by its document frequency.

TF is defined as follows:

$$TF_{i,j} = \frac{p_{i,j}}{\sum_k p_{k,j}} \tag{6}$$

where $p_{i,j}$ is the number of occurrences (*tf*) of the considered word (term) in document d_j, and the denominator is the sum of number of occurrences of all terms in document d_j.

While IDF is defined as follows:

$$IDF = \log \frac{|D|}{|\{d : w_i \in d\}|}, \tag{7}$$

where $|D|$ is cardinality of D, or the total number of documents in the training samples and $|\{d : w_i \in d\}|$ is number of documents where the term (word) w_i appears (that is $p_{i,j} \neq 0$). If the term is not in the training samples, this will lead to a division-by-zero. It is therefore common to use smoothing: $1 + |\{d : t_i \in d\}|$.

If a document to be classified is represented by a query vector $q = (\mathbf{w}_{1,q}, \mathbf{w}_{2,q}, \ldots, \mathbf{w}_{t,q})$, then distance between q with a document d_j in the training samples can be calculated using cosine similarity. Cosine similarity defines similarity (distance) between a query document with a training document. A cosine value of zero means that the query document and training document vector are orthogonal and have no match.

$$sim(d_j, q) = \frac{\mathbf{d_j} \cdot \mathbf{q}}{\|\mathbf{d_j}\| \, \|\mathbf{q}\|} = \frac{\sum_{i=1}^{N} \mathbf{w}_{i,j} * \mathbf{w}_{i,q}}{\sqrt{\sum_{i=1}^{N} \mathbf{w}_{i,j}^2} * \sqrt{\sum_{i=1}^{N} \mathbf{w}_{i,q}^2}} \tag{8}$$

In a simpler term count model, the term specific weights do not include the global parameter. Instead the weights are just the counts of term occurrences: $\mathbf{w}_{t,d} = tf_{t,d}$.

If each emotion class c is represented by a set of documents $M_c = \{d_{1,c}, d_{2,c}, \ldots, d_{|M_c|,c}\}$, then the classification result is $VSM(q) = \text{argmax}_c(\sum_{j=1}^{|M_c|} sim(d_{j,c}, q))$.

3) k-NN: Cosine similarity can be used also in k-NN technique to classify the document. In its simplest form, using $k = 1$, k-NN assigns a class of the nearest (the greatest cosine similarity value) training document as the class of a query document.

TABLE I
BASIC EMOTIONS CLASSIFICATION

Psychologist	Basic emotions
Plutchik	Anger, anticipation, trust disgust, joy, fear, sadness, surprise
Ekman, Friesen, Ellsworth	Anger, disgust, fear, joy, sadness, surprise
Frijda	Desire, happiness, interest, surprise, wonder, sorrow
Izard	Anger, contempt, disgust, distress, fear, guilt, interest, joy, shame, surprise
James	Fear, grief, love, rage
Mowrer	Pain, pleasure
Oatley and Johnson-Laird	Anger, disgust, anxiety, happiness, sadness

B. Emotions and Facial Expressions

Psychologists have tried to explain the human emotions for decades. However, they have not yet agreed upon a set of basic human emotions [20], as shown in Table I. They disagree on the exact number of affects, i.e. basic emotions, but most include 5 (five); *joy*, *sadness*, *anger*, *fear*, and *disgust* [21].

A well known model of emotions is the work of [22]. He uses basic emotions as building blocks for derived emotions; secondary emotions and even ternary emotions. All other emotions are mixed or derivative states; that is, they occur as combinations, mixtures, or compounds of the basic emotions, as depicted in Fig. 2.

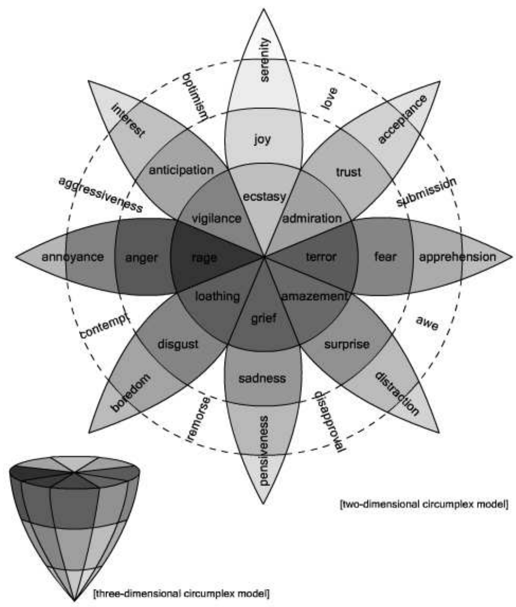

Fig. 2. Plutchik's Wheel of Emotion

[23] believed there exists a relationship between facial expression and emotional state. The proponents of the basic emotions view [24], [25], according to [26], assume that there is a small set of basic emotions that can be expressed distinctively from one another by facial expressions. For

TABLE II
FACIAL EXPRESSIONS OF BASIC EMOTIONS

No	Basic emotions	Textual description of facial expressions
1	*Joy*	The eyebrows are relaxed. The mouth is open and the mouth corners pulled back toward the ears.
2	*Sadness*	The inner eyebrows are bent upward. The eyes are slightly closed. The mouth is relaxed.
3	*Fear*	The eyebrows are raised and pulled together. The inner eyebrows are bent upward. The eyes are tense and alert.
4	*Anger*	The inner eyebrows are pulled downward and together. The eyes are wide open. The lips are pressed against each other or opened to expose the teeth.
5	*Disgust*	The eyebrows and eyelids are relaxed. The upper lip is raised and curled, often asymmetrically.
6	*Surprise*	The eyebrows are raised. The upper eyelids are wide open, the lower relaxed. The jaw is opened.

instance, when people are angry they frown and when they are happy they smile.

The six basic emotions defined by [24] can be associated with a set of facial expressions. [27] has designed a face model with 6 (six) facial expressions of basic emotions depicted in Fig. 3. Table II shows textual descriptions of facial expressions as representations of basic emotions, taken from [28]. In this paper, *surprise* is excluded due to unavailability of *surprise* class in the training set, however facial expression of *shame* is added based on following arguments from psychological researches.

Although *guilt*, *shame* and *embarrassment* are terms meant to refer to different emotions, researchers attempting to demonstrate distinctions in how people actually experience these emotions are likely to encounter challenging difficulties [29].

Specifically, even though these emotions are distinct [30], *guilt* does not have a distinct facial display [31], [32]. *Guilt* may involve a complex pattern of facial, gaze, postural, and speech activity [33] that can not be displayed merely by a static facial picture.

Further question to be answered is ”Does *shame* have distinct facial expression?” The English word ”red-faced” has a Chinese equivalent *lianhong* (literally ”face-red”). Both suggest a connection between *shame* and facial display [29]. A number of non verbal behaviour could indicate *shame*. They include hiding behaviour such as covering all or parts of the face, gaze aversion, with eyes downcast or averted, bowed head [31], hunching shoulders, squirming, fidgeting, blushing (”red-faced”), biting or licking the lips, biting the tongue, or false smiling [34]. Eyes downcast and unique mouth shape (as result from lips biting) are chosen to represent *shame* facial expression in our experiments.

Another aspect to consider is the intensity of emotion. In everyday life, emotions often occur in mixtures. For example, the feeling of *sadness* is often mixed with *shame*, *anger* or *fear*. People typically respond to social events with an admixture of emotions, with varying degrees of intensity. Plutchik’s model also explains the notion of emotion intensity, that represents the strength by which an emotion is felt. Plutchik’s model has a major advantage, because it significantly decreases the complexity of classification due to a small number of basic emotions to which the system can be restricted.

The intensity of emotion can be characterized qualitatively in daily conversation using words, such as ”little”, ”a bit”, ”rather”, ”very”, ”quite” or ”extreme” (for example ”a bit angry”, ”very happy”) and for modeling this kind of different degrees of emotion expression, fuzzy-based method is a suitable solution.

C. Computational Work on Emotional Facial Expressions

An essential element for adding a human personality in a life-like character is the facial expression. There are many attempts to create emotional facial expressions on a virtual life-like character. Facial expressions in the InterFace [35] were modelled visually and interactively. Complex one were modelled by combining a small set of pre-modelled expressions, called the Basic Library of Expressions (BLE).

Each expression stored in the BLE can have its own weight on the character model. The final expression of model was composed by adding the differences stored for each expression in the BLE, multiplying by the respective weights. Unlike our proposed system which facial expression is displayed automatically following text input, the InterFace system was designed to be operated by animator.

Emotion generation in an algebraic model developed by [36] defined a set of parameters motivators. These motivators consist of keywords (mapped to emotional facial expressions, for example ”happy” to ”smile”), hierarchical context to calculate coherence, user profile which associated with inhibitor/amplifier affecting facial display and biological needs (for example, blinking and wetting the lips). Motivators with their respective weights then were feed into a multiplexer to display a final facial expression. In our system, instead manual input of keywords, machine learning approach is employed to process the text.

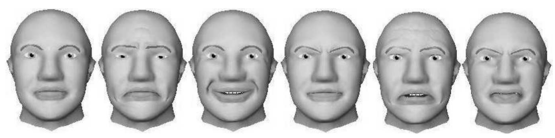

Fig. 3. Facial expression of neutral and basic emotions: *sadness*, *joy*, *anger*, *fear*, *disgust*, *surprise* (from left to right) [27].

III. SYSTEM DESIGN

Our proposed system as depicted in Fig. 4 is built from two modules: (a) basic emotions classification based on text input using Naïve Bayes supervised text classifier and (b) fuzzy-based emotion expression of a life-like character’s face.

A. Naïve Bayes Text Classifier for Emotion Classes

Probability, for a Bayesian, is a way to represent a degree of belief in an event, given the evidence. In real world applications, probability value will lie between 0 and 1. It means that text document d is categorized into a single emotion class c_i with the highest degree of belief and a

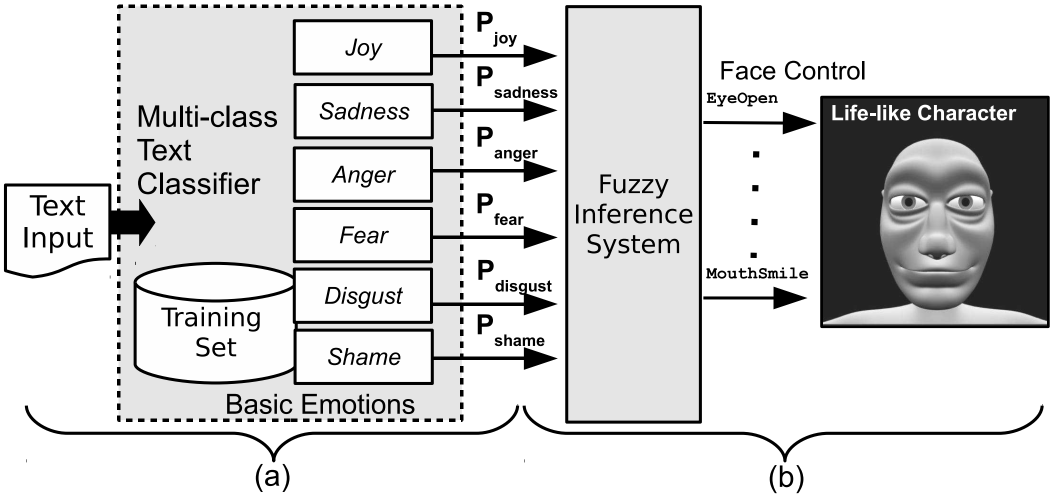

Fig. 4. Overview of our proposed system: (a) emotion text classification and (b) Fuzzy Inference System for facial expression

number of lower degree of beliefs to the other classes of emotion C. In term of basic emotion classification from text, a sentence d will generate the one "strongest" (highest probability value) basic emotion and a number of "weaker" (lower probability values) basic emotions .

The usual usage of text classification is to classify a text into one class of emotion and uses only highest value of probability to determine which class a classified text should belong to. However in our system, all of probability values resulted from NB classifier are utilized. It is assumed that probability values are to represent intensity of basic emotions. For example, zero (0) probability for P_{joy} means no basic emotion of *joy* is felt, while value 0.2 means "a bit" of *joy*. One (1) indicates maximum value of certain basic emotion. In real world applications, each basic emotion will contribute their intensities differently through their probability values as the results from a text classifier. Therefore, in our proposed system, a final facial expression will be triggered from a mixture of basic emotions.

Dataset used came from ISEAR (International Survey on Emotion Antecedents and Reactions) [37] which was conducted in 1990s across 37 countries and had almost about 3000 respondents. ISEAR dataset consists of 7,666 sentences and snippets in English, categorized into 7 (seven) classes of emotion: *joy*, *fear*, *anger*, *sadness*, *disgust*, *shame*, and *guilt*. This data set contains text documents of about 3-4 sentences pre-classified into the categories of emotion.

Due to previously mentioned reason that *guilt* can not be expressed by a simple facial picture, *guilt* is excluded and left out 6 (six) classes of emotion. Table III shows some sentences taken from ISEAR dataset with their corresponding emotions.

To measure the performance of a text classifier, cross-validation technique is used. The goal is to estimate how accurately a predictive model will perform in practice. One round of cross-validation involves partitioning a data-set into two subsets, performing the analysis on one subset (the training set), and validating the analysis on the other subset (the validation set or testing set). In our experiment, Leave-One-Out Cross-validation (LOOCV) technique is employed.

LOOCV involves using a single observation from the original sample as the validation data, and the remaining observations as the training data. This is repeated such that each observation in the sample is used once as the validation data. Although LOOCV is usually very expensive from a computational point of view because of the large number of times the training process is repeated, LOOCV is considered to be able to show the near actual performances of text classifiers in real world applications, because almost all of the training data (except one) can be used. In the actual usage, text classifier will use all of the training data and should be able to classify a new unknown input.

TABLE III
SAMPLE OF ISEAR DATASET

Emotion	Sentence
Joy	After my girlfriend had taken her exam we went to her parent's place
Sadness	My grandmother died
Fear	At the dentist's, waiting for my turn to come
Anger	Having a fight with a class mate
Disgust	When I was weeding the garden I found a lizard in my hand
Shame	Cheating to get the best grade on a test in 7th grade.

A number of methods for text classification such as SNB, MNB, CNB, VSM and k-NN are employed in our experiments and the performances of each method are evaluated using LOOCV.

B. Facial Expressions of Ludwig

In the proposed system, the probability values of basic emotion class, using fuzzy-based mechanism, control the expression of face model. Ludwig as face model is employed for our experiments. Ludwig [38] is a full body fully rigged and animation ready character for Blender. Blender[2] is a free graphics application that can be used for 3D modelling and rendering.

Ludwig has many face controls. Most of them are utilized, namely `EyesDirection`, `BrowPosition.R/L`, `BrowEmotion.R/L`, `BrowWrinkle`, `EyeOpen.R/L`,

[2] http://www.blender.org

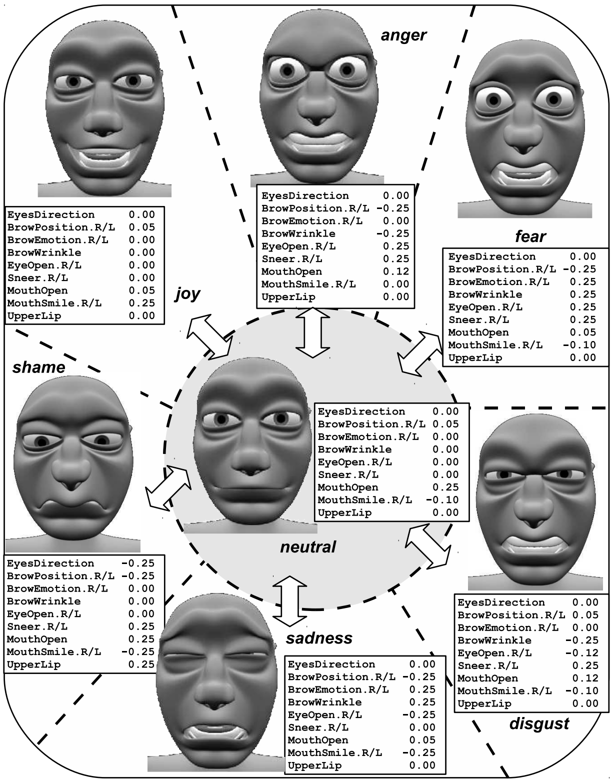

Fig. 5. Facial expressions of Ludwig in basic emotions and their corresponding face control parameters.

`Sneer.R/L`, `MouthOpen`, `MouthSmile.R/L` and `UpperLip`.

Face parameters are set by visual observations of Ludwig face for lowest and highest value for face control as ranges of operating parameter values (see Table IV). Table II is referred as guidelines to manually set facial parameters, hence generating facial expressions for 5 (five) basic emotions, while *shame* –as revealed by our investigation in previous section– is facially expressed by eyes downward and lip biting . For facial expression of *shame*, `EyesDirection` and `UpperLip` face controls are needed.

A neutral state of face or a face without emotional expression is also added. Neutral expression is represented by a very thin smile in the face, the same as [27]. Fig. 5 depict facial expressions of Ludwig in basic emotions surrounding neutral expression in the center and their corresponding face control parameters.

C. Fuzzy Inference System for Facial Expressions of Ludwig

Fuzzy inference is the process of formulating the mapping from given inputs namely probability values as results from NB text classifier, to an output which is face parameter control of life-like character, using fuzzy-based mechanism.

The process of fuzzy inference involves Membership Functions (MF), logical operations, if-then rules, aggregation and defuzzification to produce output. Fuzzy Inference System (FIS) Mamdani is implemented in java-based software jFuzzyLogic[3] which supports FCL (Fuzzy Control Language) [39] file format.

Inputs consist of 6 (six) probability values of six emotion classes. Each input has following three linguistic variables: "`low`", "`medium`" and "`high`", implemented using triangle MF.

Input is categorized as "`low`", when input value is between -0.4 and 0.4; "`medium`" when input value is between 0.1 and 0.9; and "`high`", when input value is between 0.6 and 1.4, as depicted in Fig. 6.

[3]http://jfuzzylogic.sourceforge.net/

TABLE IV
OPERATING PARAMETER VALUES OF LUDWIG'S FACE

Parameter	Lower	Upper
`EyesDirection`	-0.25 (look down)	0 (straight)
`BrowPosition.R/L`	-0.25	+0.25
`BrowEmotion.R/L`	0.0	+0.25
`BrowWrinkle`	-0.25	+0.25
`EyeOpen`	-0.25 (close)	+0.25 (open eyed)
`Sneer.R/L`	0.0 (neutral)	0.25 (sneer)
`MouthOpen`	0.0 (open)	0.25 (close)
`MouthSmile.R/L`	-0.25 (frown/down)	0.25 (smile/up)
`UpperLip`	0.00	0.25

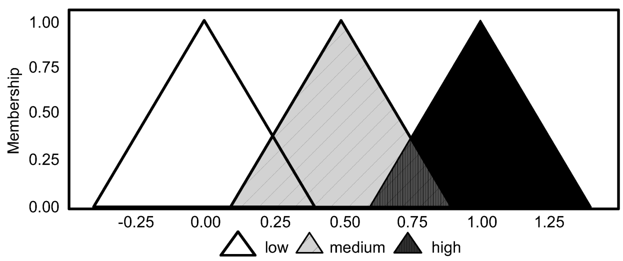

Fig. 6. Membership Function (MF) of input

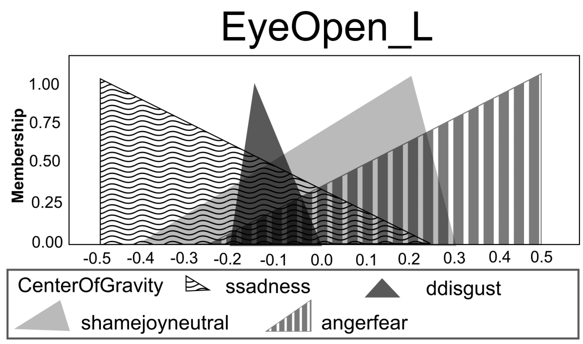

Fig. 7. MF of `EyeOpen_L` output

While for the output, essentially, logic pair "1" is assigned to all of values listed as corresponding face control parameters (see Fig. 5). To illustrate more details on how linguistic variables are assigned for an output, let's take an example left part of `EyeOpen.R/L` control (R/L stands for "Right/Left"), which is `EyeOpen_L`.

From face control parameters which are previously defined, `EyeOpen_L` control should have logic pair 1 for *sadness* at -0.25; (*disgust*) at -0.12; (*shame*, *joy*, neutral) at 0; and (*anger*, *fear*) at 0.25. Linguistic variables for MF output are "`ssadness`", "`ddisgust`", "`shamejoyneutral`" and "`angerfear`" (see Fig. 7). All are implemented using

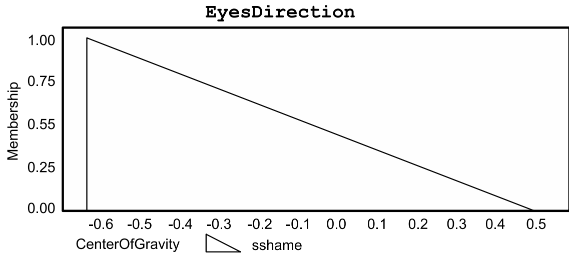

Fig. 8. Membership function of `EyesDirection` output

triangle MF and Center Of Gravity (COG) defuzzification method.

Using triangle MF designed as depicted by Fig. 7 and Center of Gravity defuzzification, if $P_{sadness} = 1$ or *sadness* feeling is at maximum state, then `EyeOpen_L` control will equal to -0.25 expressing *sadness* by closing left eye of Ludwig (see Fig. 5 *sadness*). If $P_{sadness}$ decreases then `EyeOpen_L` control will gradually move to zero (see "`ssadness`" MF triangle) causing opening of the left eye.

Assignment "(*shame*, *joy*, neutral) at 0" means for `EyeOpen_L` control, expression of *shame* shares common appearance with (*joy*, neutral) which represents by "`shamejoyneutral`" MF, while assignment "(*anger*, *fear*) at 0.25" means *anger* shares common appearance with *fear* which represents by "`angerfear`" MF.

All of linguistic variables of inputs and outputs need to be related using rules. Rules will select which MF will be used in defuzzification stage. As already shown, these relations are simple and intuitive.

For example, if probability of *sadness* is "`high`" then "`ssadness`" defuzzification map will be activated for `EyeOpen_L` control and written as "`IF (sadness IS high) THEN EyeOpen_L IS ssadness`" rule. While for "`medium`" *sadness*, the same map applied but the weight is reduced by half, hence the rule in FCL is written as "`IF (sadness IS medium) THEN EyeOpen_L IS ssadness WITH 0.5`". As a rule of thumb for our fuzzy-based mechanism, for "`medium`" MF input, the same defuzzification map for "`high`" MF input is applied, but weight is reduced by half (using "`WITH 0.5`" FCL command).

If all of probability values are "`low`", MF for displaying neutral expression will be utilized, namely "`shamejoyneutral`", then the rule is written as "`IF (disgust IS low) AND (fear IS low) AND (joy IS low) AND (sadness IS low) AND (shame IS low) AND (anger IS low) THEN EyeOpen_L IS shamejoyneutral`". Table V shows all rules for `EyeOpen_L` control. Since the right part of control is symmetry with the left part, the same fuzzy-based mechanism also applies to `EyeOpen_R` control.

FIS implementation of `EyesDirection` control is simpler. Fig. 8 depicts defuzzification mapping for `EyesDirection`. There are only two rules: when intensity of *shame* is "`high`", then "`sshame`" will be selected and if intensity of *shame* is "`medium`", then weight is set to 0.5 and same map is applied. Other than these two conditions, output is set to default (0) which represents neutral facial expression.

Triangle MF shown in Fig. 8 implies that intensity decrease of *shame* will increase value of `EyesDirection` control, which moves from -0.25 to 0, in other words, from "*shame*" to "a bit *shame*" to "neutral" emotion will cause eyes go upward from look down position to straight.

In total, there are 57 rules, covering 9 (nine) face controls, consist of `BrowPosition.R/L` 5 rules, `BrowEmotion.R/L` 5 rules, `BrowWrinkle` 7 rules, `EyeOpen.R/L` 9 rules, `Sneer.R/L` 5 rules, `MouthOpen` 7 rules, `MouthSmile.R/L` 9 rules, `UpperLip` 5 rules and `EyesDirection` 2 rules,.

IV. Experiments and Discussions

Since proposed system consists of two modules, the discussion starts with text classification and followed by facial expression of life-like character.

A. Text Classification

Simple NB (also VSM and k-NN) are implemented using C# programs and java-based programs utilise Weka [40] library for Multinomial NB and Complement NB, as experiment tools.

Original ISEAR dataset consists of 7,666 text files of 7 (seven) emotion class. After removing invalid entries, such as "[No response]", leave behind *guilt* emotion class, left out 6 (six) emotion classes which are *anger, disgust, fear, joy, sadness* and *shame* with their corresponding number of text files: 1,086; 1,078; 1,090; 1,090; 1,083 and 1,072 respectively; in total 6,499 files.

Text is pre-processed using StopWords removal, optional lemmatization and feature sets are formed based on *tf*, *tf·IDF* and *TF·IDF* using bag-of-word approach. StopWords list is manually defined, which consists of 63 words that are considered not affecting text classification, such as "where", "which", "their", "himself", "thing", "those". Python-based MontyLemmatiser module from MontyLingua [41] is employed for lemmatization.

A lemmatizer is different from a stemmer. The difference is that a stemmer operates on a single word without knowledge of the context and usually based merely on rules. For example, the inflected forms "go", "goes", "going", "went", and "gone" will be processed by a lemmatizer to the lemma "go", while a stemmer may not do the same. The stem produced by a stemmer, need not be identical to the morphological root of the word (lemma). It is usually sufficient that related words map to the same stem, even if this stem is not a valid meaningful word. Therefore, stemmers are typically easier to implement and run faster, with accuracy trade-off.

Previously, the performance of Simple NB text classifier [11] have been evaluated. Experiments were conducted using our Indonesian translated version from some portion of ISEAR dataset and showed best F1 equaled to 62.15% with split ratio 80/20.

Table VI shows our latest results using LOOCV. CNB "with lemmatization" achieves highest score (63.53% accuracy), close to score of CNB "without lemmatization" (63.44% accuracy), followed by MNB "with lemmatization" (62.47% accuracy) at third place, while Table VII displays details for each of 6 (six) emotion classification scores gained by each method.

TABLE V
RULES FOR EYEOPEN_L

RULE 1: IF (disgust IS low) AND (fear IS low) AND (joy IS low) AND (sadness IS low) AND (shame IS low) AND (anger IS low) THEN EyeOpen_L IS shamejoyneutral;
RULE 2: IF (fear IS high) OR (anger IS high) THEN EyeOpen_L IS angerfear;
RULE 3: IF (joy IS high) OR (shame IS high) THEN EyeOpen_L IS shamejoyneutral;
RULE 4: IF (sadness IS high) THEN EyeOpen_L IS ssadness;
RULE 5: IF (disgust IS high) THEN EyeOpen_L IS ddisgust;
RULE 6: IF (sadness IS medium) THEN EyeOpen_L IS ssadness WITH 0.5;
RULE 7: IF (joy IS medium) OR (shame IS medium) THEN EyeOpen_L IS shamejoyneutral WITH 0.5;
RULE 8: IF (anger IS medium) OR (fear IS medium) THEN EyeOpen_L IS angerfear WITH 0.5;
RULE 9: IF (disgust IS medium) THEN EyeOpen_L IS ddisgust with 0.5;

In general, lemmatization increases performance of each method, however careful examination from Table VII reveals that it is not the case for every emotion class. In *shame* class, lemmatization decreases performances, although insignificant for Complement NB method.

Probability values as results of text classification using MNB can be directly used as inputs for FIS. However, scores from Complement NB text classifier need to be processed further, because by its nature, CNB scores are complement values, which mean smallest value represents greatest intensity of emotion (strongest emotion). To rectify these, simple calculations for CNB scores are applied. Firstly, the original score of CNB s is inverted using reciprocal function $f(s) = 1/s$, then all of inverted scores are summed together to be used latter in normalization. As an example, an unknown sentence "Hurricane Igor gets stronger, storm Julia follows"[4] will result CNB scores $s_{anger}, s_{disgust}, s_{fear}, s_{joy}, s_{sadness}, s_{shame}$ equal to -18,032, -18,190, -18,956, -18,215, -18,356, -18,072 respectively. This sentence is classified by MNB with $P_{anger}, P_{disgust}, P_{fear}, P_{joy}, P_{sadness}, P_{shame}$ equal to 0.054, 0.133, 0.390, 0.150, 0.211, 0.063 respectively and by CNB (after inversion) equal to 0.120, 0.141, 0.303, 0.144, 0.166, 0.125 respectively. Both methods show *fear* as the strongest emotion of the sentence.

B. Facial Expression of Life-like Character

As a start of evaluation, the fuzzy-based facial expression module is tested with all of facial expressions of basic emotions as depicted in Fig. 5. Facial expression of life-like character must be the same as designed, when one of probability values of basic emotion is at maximum state and other probability values are zero. For example, if $P_{sadness} = 1.0$ and other probability values are zero, then Ludwig must display exactly the same as *sadness* facial expression as depicted in Fig. 5. Neutral expression is displayed when all of probability values are zero. All of 7 (seven) facial expressions are tested and displayed as expected.

As baseline test for facial expression, two sentences are taken from ISEAR data set. Using LOOCV (which means each sentence is excluded from training and use the rest of ISEAR data set for each experiment), Fig. 9 and 10 depict *shame* and *joy* facial expressions from their respective sentence.

[4]http://www.thejakartaglobe.com/afp/hurricane-igor-gets-stronger-storm-julia-follows/395994 Source: Agence France-Presse (AFP) Date: 13 Sep 2010

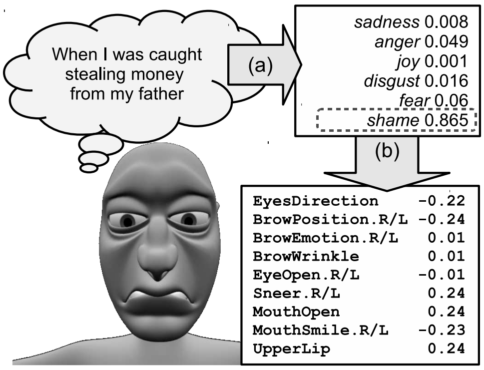

Fig. 9. A *shame* facial expression with corresponding sentence using LOOCV.

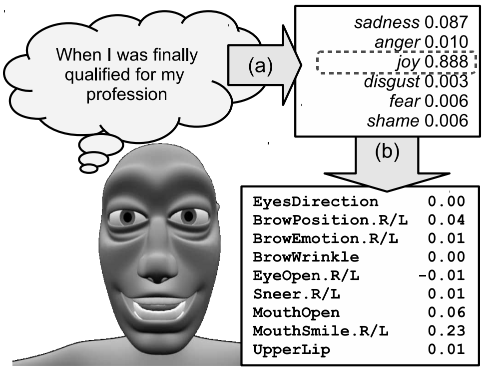

Fig. 10. A *joy* facial expression with corresponding sentence using LOOCV.

For example of real world application, two sentence are taken from Internet news to be processed by the system. Fig. 11 and 12 show facial expressions with their corresponding text input. As shown in Fig. 11, facial expression of Ludwig displays not only strongest emotion, but a mixture of basic emotions. This non-animated facial expression of Ludwig can be imagined as a single spontaneous reaction immediately after receiving the text input.

Previously using similar approach implemented in Matlab FIS, [42] have conducted a small scale survey by asking human viewers about the appropriateness of generated facial expression from 5 (five) basic emotions, consisted of *joy*, *sadness*, *anger*, *fear*, and *disgust*. The data used were Indonesian translation from English ISEAR data-set, not entire data-set tough, but only a fraction. Twenty facial expressions of Ludwig along with 20 (twenty) corresponding Indonesian sentences, were showed to 100 (one hundred) respondents. They should choose one from two possible answers; "Yes,

TABLE VI
EXPERIMENTAL RESULTS USING LOOCV FOR 6 (SIX) CLASS EMOTION CLASSIFICATION.

DataSet	Method	Mean Recall (%)	Mean Precision (%)	Mean Specifity (%)	Mean F1 (%)	Accuracy (%)
ISEAR	k-NN *tf·IDF*	42.37	41.91	78.66	41.82	42.41
	k-NN *TF·IDF*	42.58	42.14	78.81	42.02	42.62
	VSM *tf·IDF*	58.78	59.56	87.79	58.53	58.82
	VSM *TF·IDF*	58.83	59.57	87.81	58.58	58.87
	SNB *tf*	54.01	54.60	85.65	54.24	54.04
	MNB	62.12	62.10	89.19	62.07	62.15
	CNB	63.40	63.35	89.67	63.12	63.44
ISEAR with lemmatization	k-NN *TF·IDF*	44.01	44.21	79.74	44.08	44.05
	VSM *TF·IDF*	60.08	60.34	88.33	60.16	60.12
	MNB	62.44	62.60	89.34	62.47	62.47
	CNB	**63.49**	**63.45**	**89.71**	**63.22**	**63.53**

TABLE VII
RESULTS OF 6 (SIX) CLASS EMOTION CLASSIFICATION.

Emotion Class	Method	ISEAR			ISEAR with lemmatization		
		Precision (%)	Recall (%)	F1 (%)	Precision (%)	Recall (%)	F1 (%)
Anger	k-NN *TF·IDF*	32.25	27.44	**29.65**	27.66	29.65	28.62
	VSM *TF·IDF*	52.39	49.36	**50.83**	44.33	47.51	45.87
	MNB	52.79	53.22	53.00	52.49	56.26	**54.31**
	CNB	54.38	52.58	53.46	57.44	55.80	**56.61**
Disgust	k-NN *TF·IDF*	43.31	34.51	**38.41**	39.74	36.46	38.03
	VSM *TF·IDF*	69.89	48.89	**57.53**	60.06	55.10	57.47
	MNB	64.24	59.00	61.51	64.91	59.55	**62.12**
	CNB	65.81	61.97	**63.83**	66.10	60.95	63.42
Fear	k-NN *TF·IDF*	49.66	47.34	48.47	50.90	51.83	**51.36**
	VSM *TF·IDF*	66.97	66.97	**66.97**	66.04	67.25	66.64
	MNB	69.88	69.17	69.53	69.55	70.83	**70.18**
	CNB	66.64	75.69	**70.88**	65.25	76.33	70.36
Joy	k-NN *TF·IDF*	49.74	61.47	54.99	59.28	61.56	**60.40**
	VSM *TF·IDF*	59.93	73.39	65.98	72.17	74.95	**73.54**
	MNB	67.71	73.67	70.56	69.35	72.02	**70.66**
	CNB	63.39	76.24	69.22	64.09	75.32	**69.25**
Sadness	k-NN *TF·IDF*	40.36	50.23	44.76	56.53	52.72	**54.56**
	VSM *TF·IDF*	51.85	65.84	58.01	71.19	66.39	**68.71**
	MNB	63.73	63.43	63.58	65.94	61.50	**63.64**
	CNB	69.30	63.16	**66.09**	68.58	61.68	64.95
Shame	k-NN *TF·IDF*	37.23	33.58	**35.31**	31.17	31.81	31.49
	VSM *TF·IDF*	56.40	48.51	**52.16**	48.26	49.25	48.75
	MNB	54.25	54.20	**54.22**	53.38	54.48	53.92
	CNB	60.58	50.75	**55.23**	59.24	50.84	54.72

it is an appropriate expression” or ”No”. After survey, total number of ”Yes” answer was 1,328 and ”No” answer was 672, equaled to 66.4% accuracy.

In real world application such as animation movie or game, the life-like virtual character of proposed system needs to be animated. Simple animation of facial expression will be transition from neutral to *sadness*, neutral to *anger* or reverse directions, *joy* to neutral, *fear* to neutral, and so on. Life-like character can also receive two (or more) consecutive text inputs. For example, if the life-like character reads one news which somehow generates ”slightly *sad*” and a moment later receives a ”quite *happy*” news, there will be a change of facial expression from one emotion to another.

The change from one type of emotion to another can be described as a transitional one type of emotion, toward neutral and then followed by transition from neutral, toward another emotion. For example, changes in emotion ”slightly *sad*” to ”quite *happy*” can be portrayed as transition from ”slightly *sad*” to ”neutral” to ”quite *happy*”. Transition mechanism of emotion with neutral expression as central is illustrated in Fig. 5.

Let say ”slightly *sad*” is represented by $P_{sadness} = 0.2$ and ”quite *happy*” is represented by $P_{joy} = 0.7$, then transition of values will be as follows: $P_{sadness}$ gradually decreases to zero (neutral) and after neutral expression is reached, P_{joy} will raise from zero to 0.7.

Fig. 13 depicts a bit more complex emotion transition. This situation is common, when Ludwig as a virtual chat friend responds to a story told by a human user, regarding ups and downs of his/her friendship. Ludwig listens to first sentence ”When friends try to put me down or hurt me” and later on receives second text ”When I met friends I had not seen for the last 2-4 years.” For the sake of clarity, these graphics are illustrated in 200 steps (frames).

These two sentences will generate two different kind of emotions, with probability values of $P_{anger}, P_{disgust}, P_{fear}, P_{joy}, P_{sadness}, P_{shame}$ equal to 0.431, 0.042, 0.028, 0.001, 0.265, 0.232 and 0.003, 0.003,

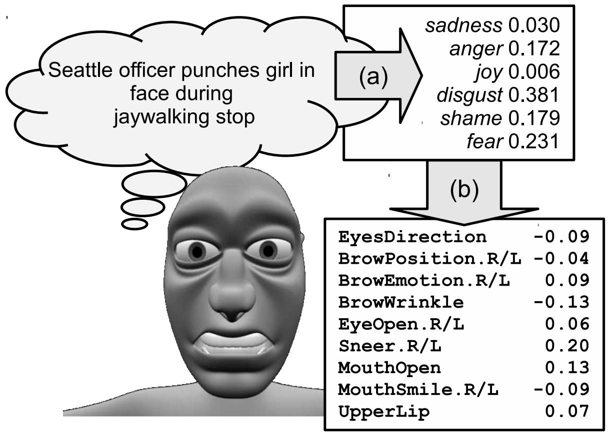

Fig. 11. Facial expressions of emotions blending between *disgust*, *fear*, *anger* and *shame*.

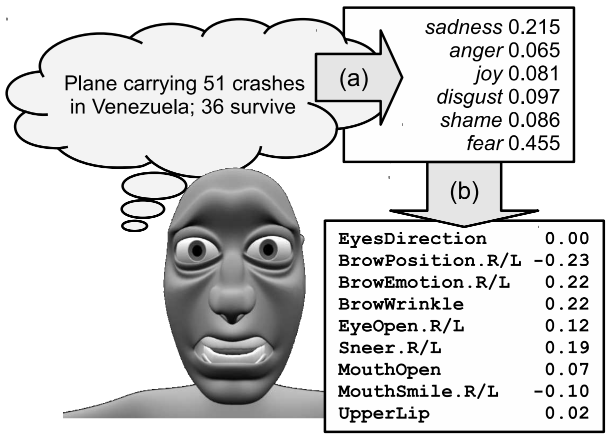

Fig. 12. Facial expressions from reading/hearing an accident news.

0.009, 0.915, 0.048, 0.022 respectively. First sentence results a mixture of emotions between *anger*, *sadness* and *shame*, while second sentence is classified as *joy*.

Fig. 13 (a) depicts gradual decrease from first emotion toward neutral, followed by gradual increase from neutral toward second emotion. Fig. 13 (b) shows the dynamics of face control values and Fig. 13 (c) displays partial samples of generated facial animation. Although there are a number of sharp turns can be seen in transition of face controls signal (Fig. 13 (b)), in the generated facial animation they are hardly noticeable to a casual observer (with the exception of `EyesDirection` signal at frame 58-59 which raises from -0.066 to 0.0). These experiments demonstrate how the life-like character will express its emotion automatically, when receiving text input, in real world application.

V. Conclusion and Future Work

Using a supervised machine learning text classifier such as Naïve Bayes method and its variants like SNB, MNB, CNB, combined with Fuzzy Inference System Mamdani, emotion representations can be displayed in the form of facial expressions of a character model as a life-like response from a text input. The proposed system is capable of displaying a blending mixture of emotions and animated facial expressions can be generated automatically.

In the future, the performance of text classifier will be improved possibly with help from another method such as semantic tool. There are some readily available semantic tools provided by NLTK [43], WordNet [44] and ConceptNet [45], need to be explored.

References

[1] M. S. El-Nasr, "Modeling emotion dynamics in intelligent agents," *M.S. thesis*, May 1988.
[2] M. Minsky, *The Society of Mind*. Simon and Schuster, 1985.
[3] J. Minato, D. B. Bracewell, F. Ren, and S. Kuroiwa, "Japanese emotion corpus analysis and its use for automatic emotion word identification," *Engineering Letters*, vol. 16, no. 1, pp. 172–177, March 2008.
[4] J. Bates, "The role of emotion in believable agents," *Communications of the ACM*, vol. 37, pp. 122–125, 1994.
[5] N. O. Andrews and E. A. Fox, "Recent developments in document clustering." *Technical Report TR-07-35*, 2007. [Online]. Available: http://eprints.cs.vt.edu/archive/00001000/
[6] F. Sebastiani and C. N. D. Ricerche, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, pp. 1–47, 2002.
[7] W. W. Cohen, "Learning rules that classify e-mail," in *In Papers from the AAAI Spring Symposium on Machine Learning in Information Access*. AAAI Press, 1996, pp. 18–25.
[8] S. Tiun, R. Abdullah, and T. E. Kong, "Automatic topic identification using ontology hierarchy," in *In the Proc. of the 2 nd International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2001)*, 2001, pp. 444–453.
[9] Y. Yang, S. Slattery, and R. Ghani, "A study of approaches to hypertext categorization," *Journal of Intelligent Information Systems*, vol. 18, pp. 219–241, 2002.
[10] T. Danisman and A. Alpkocak, "Feeler: Emotion classification of text using vector space model," in *AISB 2008 Convention, Communication, Interaction and Social Intelligence*, vol. vol. 2, Aberdeen, UK, April 2008.
[11] I. Destuardi and S. Sumpeno, "Klasifikasi emosi untuk teks bahasa indonesia menggunakan metode naive bayes," in *Seminar Nasional Pascasarjana*. Institut Teknologi Sepuluh Nopember (ITS) Surabaya, August 2008.
[12] D. Lewis, "Naive (bayes) at forty: The independence assumption in information retrieval," in *Machine Learning: ECML-98*, ser. Lecture Notes in Computer Science, C. Nédellec and C. Rouveirol, Eds. Springer Berlin / Heidelberg, 1998, vol. 1398, pp. 4–15, 10.1007/BFb0026666. [Online]. Available: http://dx.doi.org/10.1007/BFb0026666
[13] A. Mccallum, "A comparison of event models for naive bayes text classification," in *In AAAI-98 Workshop on Learning for Text Categorization*. AAAI Press, 1998, pp. 41–48.
[14] J. D. M. Rennie, L. Shih, J. Teevan, and D. R. Karger, "Tackling the Poor Assumptions of Naive Bayes Text Classifiers," *Proceedings of the Twentieth International Conference on Machine Learning*, 2003.
[15] Y. Yang, "An evaluation of statistical approaches to text categorization," *Journal of Information Retrieval*, vol. 1, pp. 67–88, 1999.
[16] T. Joachims, "Text categorization with support vector machines: learning with many relevant features," in *Proceedings of ECML-98, 10th European Conference on Machine Learning*, C. Nédellec and C. Rouveirol, Eds. Heidelberg et al.: Springer, 1998, pp. 137–142. [Online]. Available: http://citeseer.ist.psu.edu/joachims97text.html
[17] R. E. Schapire and Y. Singer, "Boostexter: a boosting-based system for text categorization," in *Machine Learning*, 2000, pp. 135–168.
[18] W. W. Cohen and Y. Singer, "Context-sensitive learning methods for text categorization," in *ACM Transactions on Information Systems*. ACM Press, 1996, pp. 307–315.
[19] K. Nigam, "Using maximum entropy for text classification," in *In IJCAI-99 Workshop on Machine Learning for Information Filtering*, 1999, pp. 61–67.
[20] A. Ortony and T. J. Turner, "What's basic about basic emotions?" *Psychological Review*, vol. 97, no. 3, pp. 315–331, 1990.
[21] P. Ekman, "An argument for basic emotions," *Cognition & Emotion*, vol. 6, no. 3, pp. 169–200, 1992. [Online]. Available: http://dx.doi.org/10.1080/02699939208411068
[22] R. Plutchik, *The Emotions*. University Press of America,Inc., July 1991.
[23] P. Ekman, "Facial expression and emotion," *American Psychologist*, vol. 48, pp. 384–392, 1993.
[24] ——, *Emotion in the Human Face*, 2nd ed. Cambridge University Press, 1982.
[25] C. E. Izard, "Emotions and facial expressions: A perspective from differential emotions theory," in *The Psychology of Facial Expression*, J. A. Russell and J. M. F. Dols, Eds. Maison des Sciences de l'Homme and Cambridge University Press, 1997.

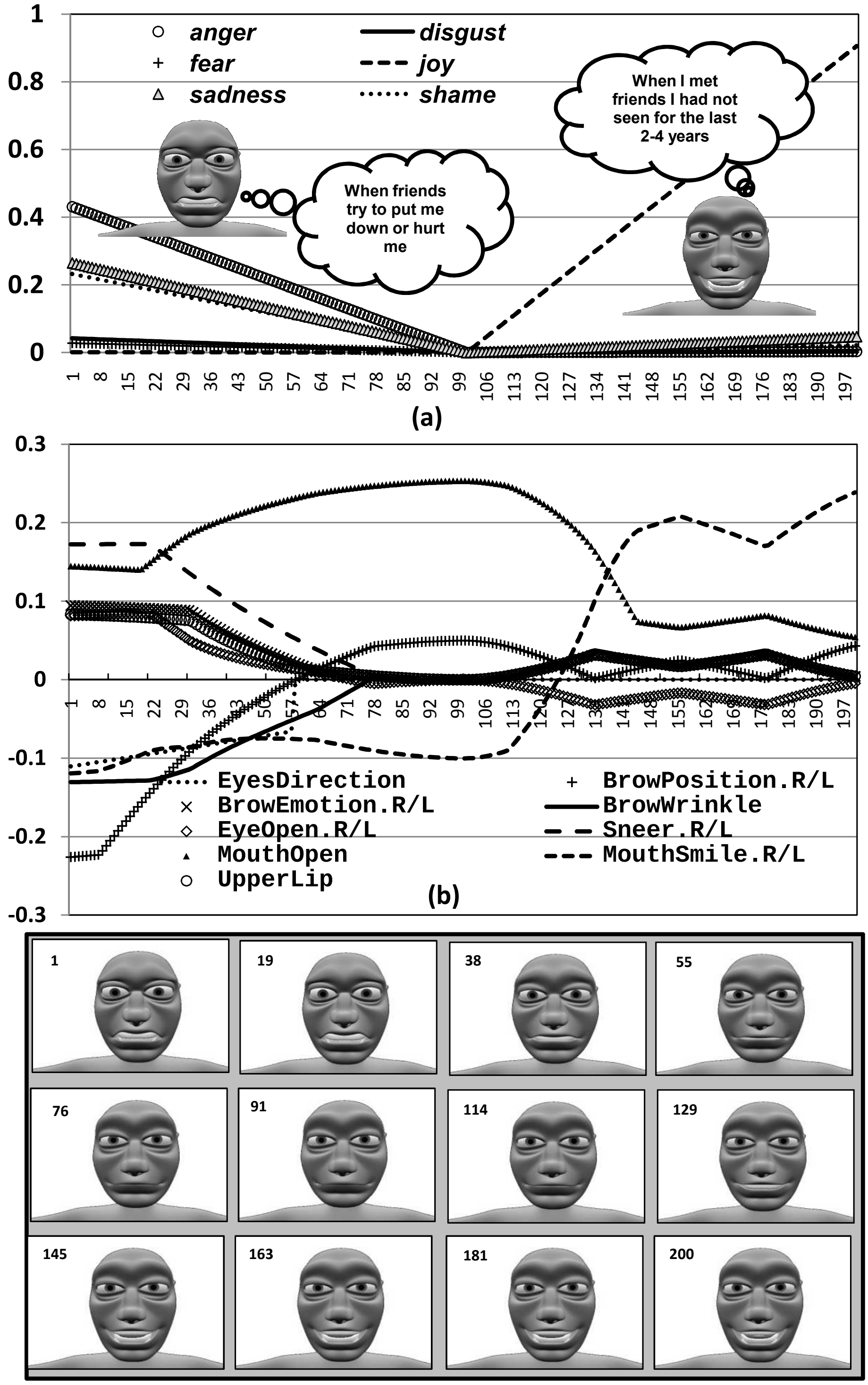

Fig. 13. Animated facial expression of emotion transition (a) Emotion transition (b) Face controls transition (c) Facial animation samples

[26] A. Kappas, “What facial activity can and cannot tell us about emotions,” in *The human face: Measurement and meaning*, M. Katsikitis, Ed. Kluwer Academic Publishers, 2003, pp. 215–234.

[27] T. D. Bui, “Creating emotions and facial expressions for embodied agents,” Ph.D. dissertation, Enschede, July 2004. [Online]. Available: http://doc.utwente.nl/41553/

[28] A. M. Tekalp and J. Ostermann, “Face and 2-d mesh animation in mpeg-4,” *Signal Processing: Image Communication*, vol. 15, no. 4-5, pp. 387–421, 2000.

[29] D. Y.-F. Ho, W. Fu, and S. Ng, “Guilt, shame and embarrassment: Revelations of face and self,” *Culture & Psychology*, vol. 10, pp. 64–84, 2004.

[30] T. JP, M. RS, F. L, and B. DH., “Are shame, guilt, and embarrassment distinct emotions?” *Journal of personality and social psychology*, vol. 70, no. 6, pp. 1256–1269, 1996. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/8667166

[31] D. Keltner and B. N. Buswell, “Evidence for the distinctness of embarrassment, shame, and guilt: A study of recalled antecedents and facial expressions of emotion,” *Cognition and Emotion*, vol. 10, pp. 155–171, 1996.

[32] C. Creed and R. Beale, “Simulated emotion in affective embodied agents,” in *Affect and Emotion in Human-Computer Interaction: From Theory to Applications*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 163–174.

[33] P. Ekman, M. O’Sullivan, W. V. Friesen, and K. R. Scherer, “Invited article: Face, voice, and body in detecting deceit,” *Journal of Nonverbal Behavior*, vol. 15, pp. 125–135, 1991, 10.1007/BF00998267. [Online]. Available: http://dx.doi.org/10.1007/BF00998267

[34] S. M. Retzinger, "Identifying shame and anger in discourse," *American Behavioral Scientist*, vol. 38, no. 8, pp. 1104–1113, 1995.
[35] J. Wey and J. Zuffo, "Interface: A real time facial animation system," in *SIBGRAPHI '98: Proceedings of the International Symposium on Computer Graphics, Image Processing, and Vision*. Washington, DC, USA: IEEE Computer Society, 1998, p. 200.
[36] A. Paradiso and M. L'Abbate, "A model for the generation and combination of emotional expressions," in *Workshop on Representing, Annotating, and Evaluating Non-Verbal and Verbal Communicative Acts to Achieve Contextual Embodied Agents, Fifth International Conference on Autonomous Agents, May 29, 2001*. McGraw-Hill, 2001.
[37] K. R. Scherer and H. Wallbott, "International survey on emotion antecedents and reactions (isear)," 1990's. [Online]. Available: http://www.unige.ch/fapse/emotion/databanks/isear.html
[38] J. Pierce, "Ludwig: Character and rig v1.0," 2005. [Online]. Available: http://jasonpierce.animadillo.com/resource/ludwig/ludwig.html
[39] T. C. N. I. E. Commission, "Iec 1131 - programmable controllers," Tech. Rep., 1997.
[40] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10–18, 2009. [Online]. Available: http://dx.doi.org/10.1145/1656274.1656278
[41] H. Liu, "Montylingua: An end-to-end natural language processor with common sense." Available at: http://web.media.mit.edu/˜hugo/montylingua., 2004.
[42] M. I. Wardhana, "Ekspresi emosi pada model wajah tiga dimensi menggunakan naive bayes dan logika fuzzy," Master's thesis, Graduate School of Electrical Engineering Department, Institut Teknologi Sepuluh Nopember (ITS), Surabaya, Indonesia, 2010.
[43] E. Loper and S. Bird, "Nltk: The natural language toolkit," May 2002. [Online]. Available: http://arxiv.org/abs/cs/0205028
[44] Fellbaum, *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press, May 1998. [Online]. Available: http://wordnet.princeton.edu/
[45] H. Liu and P. Singh, "Conceptnet — a practical commonsense reasoning tool-kit," *BT Technology Journal*, vol. 22, no. 4, pp. 211–226, 2004.

Surya Sumpeno graduated from Electrical Engineering, Institut Teknologi Sepuluh Nopember (ITS) Surabaya, Indonesia for his bachelor in 1996 and received MSc. degree in 2007 from Graduate School of Information Science (GSIS) Tohoku University, Japan. Since 2007, he has been studying at Graduate School of Electrical Engineering ITS Surabaya Indonesia as a doctoral student.

He works as a lecturer of Electrical Engineering Dept ITS. His research interest include natural language processing, human computer interaction and artificial intelligence. He is IAENG member and IEEE student member.

Mochamad Hariadi graduated from Electrical Engineering Institut Teknologi Sepuluh Nopember (ITS) Surabaya, Indonesia, for his bachelor in 1995 and received MSc and PhD degree in 2003 and 2006 respectively from Graduate School of Information Science (GSIS) Tohoku University, Japan.

He is currently lecturer of Electrical Engineering Dept ITS. His research interest include multimedia processing and artificial intelligence. He is IEEE member and IEICE member.

Mauridhi Hery Purnomo received his bachelor degree from Institut Teknologi Sepuluh Nopember (ITS), Surabaya, Indonesia in 1985. He received M.Eng., and Ph.D degrees from Osaka City University, Osaka, Japan in 1995, and 1997, respectively. He has joined ITS in 1985 and has been a Professor since 2004. His current interests include intelligent system applications, electric power systems operation, control and management. He is a IEEE Member.