Counseling Request Classifying Method using Self Organizing Map

Kazuya Mera, and Takumi Ichimura

Abstract—Recently, our life style has two kinds of communities, one is a real face-to-face communication and the other is virtual communication. In school SNS, counseling system for university students has been developed. When the system receives a request for counseling, the scheduling function assigns the counseling appointment to an appropriate counselor by the adjustment of schedules. Simultaneously, the system analyzes the contents of counseling requests from the students to classify them into three annoying patterns {living, study, and job} by morphological properties. This method analyzes the distribution of morphemes in the input text by using Self Organizing Map. The system can show the analyzing result of annoying pattern of the counseling request and its emotional degree for each category by Emotion Generation Calculation.

Index Terms—School Counseling, Self Organizing Map, Morphological Analysis, Feature Word, Annoying Pattern, Emotion Generation Calculation

I. INTRODUCTION

RECENTLY, we may meet virtual human communication where the relations among human society are constructed in the Internet as seen in SNS (Social Network Services)[1]. The circumstance in our life formulates 2 kinds of communications; real face-to-face communication in daily life and virtual communications in the Internet. In Japan, SNS has been developed in the field of open source software, and some small communities by SNS are utilized to make high activities in their local communities. School SNS has also been provided some scenes to reinvigorate the student’s activities such as studies, club, friendship among their communities, course in a future, alumni association, and so on. Moreover, their activities and thinking are changing according to their grade and course. However, the community in SNS is a virtual communication space in which there consists of the structure to reflect the interpersonal relationship in the school life, and it is different from a real communication one where the user’s idea and feeling are reflected as it is. Counseling for mental care is required in not only our daily school life but also virtual communication.

...
request from the students into the three types of annoying patterns by using SOM which is trained by the morphological properties of texts of counseling contents. The system sends an e-mail to the corresponding tutor with the counseling schedule and classification result.

1. I feel downhearted, blue, and sad.
2. Morning is when I feel the best.
3. I have crying spells or feel like it.
4. I have trouble sleeping through the night.
5. I eat as much as I used to.
6. I enjoy looking at, talking to, and being with attractive women/men.
7. I notice that I am losing weight.
8. I have trouble with constipation.
9. My heart beats faster than usual.
10. I get tired for no reason.
11. My mind is as clear as it used to be.
12. I find it easy to do the things I used to do.
13. I am restless and can’t keep still.
15. I am more irritable than usual.
16. I find it easy to make decisions.
17. I feel that I am useful and needed.
18. My life is pretty full.
19. I feel that others would be better off if I were dead.
20. I still enjoy the things I used to do.

In order to check the depression rate of the student by himself/herself, we adopt “Zung Self-rating Depression Scale: SDS” [3]. It contains 20 items, with 10 items keyed negatively and 10 positively as shown in Figure 2. For each item, the subject rates whether the item occurred 1 = “a little of the time,” 2 = “some of the time,” 3 = “a good part of the time,” or 4 = “most of the time.” To obtain a total severity score, positive items are reversed, and then all items are summed. SDS scores are interpreted as follows: 23-47, within normal range; 39-59, minimal to mild depression; 53-67, moderate to severe depression as shown in Figure 3. Our system recommends consulting a doctor when the score is over 53 because our system tries to judge severe depression.

III. COUNSELING REQUEST CLASSIFYING METHOD

Our proposed system can classify the counseling requests from the students into three patterns; living, study and job using Self Organizing Map which learned grammatical features in the counseling requests.

A. Self Organizing Map

The basic SOM [4] can be visualized as a sheet-like neural network array as shown in Figure 4, the cells (or nodes) of which become specifically tuned to various input signal patterns or classes of patterns in an orderly fashion. The learning process is competitive and unsupervised, which means that no teacher is required to define the correct output for an input. Only one map node called a winner node at a time is activated corresponding to each input. The map consists of a regular grid of processing units. A model of some multidimensional observations, eventually a vector consisting of features, is associated with each unit. The map attempts to represent all the available observations with optimal accuracy using a restricted set of models. At the same time the models become ordered on the grid so that similar models are close to each other and dissimilar models are far from each other.

A sequential regression process usually carries out fitting to the model vectors. The number of input signals. An input vector is compared with all the model vectors . The best-match unit on the map is identified. The unit is called the winner. For each sample , first the winner index is identified by the condition.
\[ \forall i, \|x - m_i\| \leq \|x - m_c\| \] (1)

After that, all model vectors or a subset of them that belong to nodes centered around node \( c \) are updated at time \( t \) as

\[
\begin{align*}
  m_i(t+1) &= m_i(t) + h_c(s(t) - m_i(t)) \quad \text{for } \forall i \in N_c(t) \\
  m_i(t+1) &= m_i(t) \quad \text{otherwise}
\end{align*}
\] (2)

Here \( h_c(\cdot) \) is the neighborhood function, a decreasing function of the distance between the \( i \)th and \( c \)th nodes on map grid. The \( N_c(t) \) specifies the neighborhood around the winner in the map array. This regression is usually reiterated over the available samples.

At the beginning of the learning process, the radius of the neighborhood is large and the range of radius becomes small according to the convergence state of learning. That is, as the radius gets smaller, the local correction of the model vectors in the map will be more specific. The \( h_c(\cdot) \) also decrease during learning.

### B. Training of SOM

In our proposed method, firstly, SOM learns the text of the various counseling contents collected from the Internet. Then, multi-dimensional vector to input into SOM is generated from text data included in the counseling content, and we classify the counseling content into three types of annoying patterns based on the label of the winner node. Each dimension in the multi-dimensional vector corresponds to the number of the valid words which can classify annoying pattern as described in the following 1). In order to pick up the valid words for classifying the patterns, the examples of the text data in the counseling contents are collected from some web based bulletin board system and some relations between the frequency of appearance of the words and the annoying patterns are found. Then we propose a method to transform the text data in the counseling content to the multi-dimensional vector.

#### 1) Defining Dimension Words

When a multi-dimensional vector is transformed from the input text data, note that the specified words, such as “Kanojo” (girl friend), “Sinkyu” (promotion), and “Naitei” (informal decision) , feature as dimension axis representing each annoying pattern in the multi-dimensional space. First, such “feature words” should be selected.

We retrieved the text data about annoying appeals, complaints, and counseling requests from some web based bulletin board system which contain the words such as “Nayami” (annoyance), “Soudan” (counseling), “Gakusei” (students), “Kurashi/Manabi/Sigoto” (living/study/job), and so on. There are various topics in the collected examples and the frequency of word appearance is different among the topics. There are 23 topics in the collected examples and we picked up a few examples for each topic. Table 1 shows all of topics in the collected text data.

The chosen examples were analyzed morphologically and each word was transformed into the original form. Then, we define 831 words which are nouns, verbs, adjectives in the examples as “feature words” for the SOM. There are some feature words defined in Table 2.

### Table I

<table>
<thead>
<tr>
<th>TOPICS IN ANOYANCE PATTERN</th>
<th>Pattern</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Living</td>
<td>romance, club activities, bully, family, university living, fashion, part time job, mental problem</td>
<td></td>
</tr>
<tr>
<td>Study</td>
<td>graduate research, atmosphere of seminar or class, interest in study, remaining in the same grade, dropout, absent from school, how to study, studying abroad, university transfer</td>
<td></td>
</tr>
<tr>
<td>Job</td>
<td>employment, job hunting, qualification, interview, internship, analysis of the self</td>
<td></td>
</tr>
</tbody>
</table>

### Table II

<table>
<thead>
<tr>
<th>FEATURE WORDS (PARTLY)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Senpai (senior), Taitei (figure), Hisshuu (compulsory subject), Tan-i (credit), Gakuhu (Faculty), Sinkyu (promotion), Naitei (informal decision), Ryokou (travel), Rirekisho (curriculum vitae), Hitomisiri (shy)</td>
</tr>
</tbody>
</table>

#### 2) Collecting Learning Data

We collected 30 examples for each annoying patterns (living, study, and job) from the web based bulletin boards to train the SOM. An average of 8.7 sentences and 295.4 characters exists in each training data. There are three very short learning data which contains less than 50 characters.

#### 3) Generating Multi-Dimensional Vector

In order to input the collected text data into the SOM, the data have to be transformed into a multi-dimensional vector form. The value of each dimensional axis is calculated from the number of the feature words in the data. The process to generate multi-dimensional vector from input data is described as follows. At first, the input text data is analyzed morphologically. Mecab [5] is used as a morphological analyzer. Next, we count up the appearance of each 831 feature words in the input text data. Word matching is done in the original form. Then, an 831-dimensional vector based on the number of feature words is generated. At last, the annoying pattern label is added to the vector by hand.

Collected 90 input text data are transformed into 831-dimensional vector and they are trained by SOM.

#### 4) Map Training by SOM Algorithm

The trained map is constructed by using input data vectors as described above subsection 3). The SOM program labels the map units according to the input data vectors. Each unit receives the labels (living, study, and job) of all the data vectors for which it is the best matching unit. The map units are then labeled according to the majority of labels “hitting” a particular map unit. The no “hit” units are left unlabeled.

1399
Figure 5 shows a labeled map trained by the input data vectors. We can see that the symbols, “Si4,” “Ma1,” and so on, indicated in the figure are the label names. “Ku” indicates “Kurashi (living),” “Ma” indicates “Manabi (study),” and “Si” indicates “Sigoto (job)” and the following number is a serial number of input data. In this figure, we can see a cluster of “Si” unit at the top-middle position and there is another cluster of “Si” at the middle-left position. And at the right side of the cluster, there is a cluster of “Ma.” We can see a “Ku” cluster at the middle-right position. Three types of labels appear at top-right. Some units have multiple winner labels. Because this map is used to output an annoying pattern for an input, a unit should have only one type of annoying pattern label. Therefore, when a unit has all the same type of annoying pattern labels or one type of label occupies the unit more than 50%, we give the unit the major label. When several types of labels share the node (i.e. no one label can occupy 50%), the label of the node becomes “unknown.”

C. Counseling Request Classifying Process

In order to classify test data (text data in input counseling request) into three annoying patterns, firstly, we analyze the data morphologically and generate 831-dimensional vector same as the process described in Section 3.B.1. Next, the system applies the vector into the trained map and calculates the winner node. Then, the system picks up the label of the winner node and outputs the label’s name as the “classify result.” One of the “living,” “study,” “job,” “unknown,” and “unlabeled” is outputted.

IV. EMOTION GENERATION CALCULATION

The text data in the message contents from students are analyzed the emotion to comprehension and grasp the kinds of depressions. In order to analyze the emotion in the text data, EGC has been developed[6-10]. This section summarize the basic idea of EGC method.

A. Emotion Generation Calculations

We can usually categorize human feelings into “pleasure,” “sadness,” “angry,” “expectation,” and so on. Such these feelings are due to discriminate whether the event is pleasure/displeasure/unknown to the event when an input is given in the form of the case frame representation. In this paper, we define the equations of Emotion Generating Calculations according to a type of case frame respectively, as shown in the table 8. In the table 3, the f’s are defined in the following.

<table>
<thead>
<tr>
<th>Type</th>
<th>Event type</th>
<th>EmotionGeneratingCalculation</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>V(S)</td>
<td>f_s × f_p</td>
</tr>
<tr>
<td>II</td>
<td>V(S,OF)</td>
<td>f_s × (f_O - f_OF) × f_p</td>
</tr>
<tr>
<td>III</td>
<td>V(S,OT)</td>
<td>f_s × (f_O - f_OT) × f_p</td>
</tr>
<tr>
<td>IV</td>
<td>V(S,OM)</td>
<td>f_s × f_OM × f_p</td>
</tr>
<tr>
<td>V</td>
<td>V(S,OS)</td>
<td>(f_S - f_OS) × f_p</td>
</tr>
<tr>
<td>VI</td>
<td>V(S,O)</td>
<td>a) f_s × (f_O × f_p)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b) f_O × f_p</td>
</tr>
<tr>
<td>VII</td>
<td>V(S,O,OF)</td>
<td>f_s × (f_O - f_OF) × f_p</td>
</tr>
<tr>
<td>VIII</td>
<td>V(S,O,OT)</td>
<td>f_s × (f_O - f_OT) × f_p</td>
</tr>
<tr>
<td>IX</td>
<td>V(S,O,OM)</td>
<td>f_s × f_O × f_p</td>
</tr>
<tr>
<td>X</td>
<td>V(S,O,1)</td>
<td>f_O × f_p × f_1</td>
</tr>
<tr>
<td>XI</td>
<td>V(S,O,OC)</td>
<td>f_O × f_OC</td>
</tr>
<tr>
<td>XII</td>
<td>Others</td>
<td>-----------------------------</td>
</tr>
</tbody>
</table>

Table III. EVENT TYPE AND EGCs
Let us give an example calculation.

Event: "Ryota dates with Kiriko."
Predicate (P) = "dates with": +0.5
Subject (S)  = "Ryota": +1.0
Object Mutual (OM) = "Kiriko": 0.0

Event type: "date with" → V(S, OM)

motion Value = \( f_S(f'_l) \times f_{OM}(Kiriko) \times f_P(dates \ with) \)
= \((+1) \times (+0.1) \times (+0.5)\)
= +0.05 → positive number (a little pleasure)

The result shows that the agent feels a little pleasure about the event "Ryota dates with Kiriko."

**B. Favorite Values**

The Favorite Value is the degree of like/dislike that the agent has for a certain object contained in an event. The value has two types: a predefined value for a predicate of an event, and a knowledge-dependent value for a case element. Favorite Value for an element varies to be correspondent with knowledge the agent has about an object. The value naturally increases when the agent feels to prefer an object or to be useful. It decreases, on the contrary, when the agent doesn’t feel so. The Favorite Value for a predicate of an event is assigned a pre-determined numerical value in this paper. In our approach, the Favorite Value for an object is calculated by extracting such situations, when the agent meets conditions where is influenced its Favorite Values by agent's emotion variations. Such situations are called Favorite Value Changing Situations, and are defined with the following three rules.

**C. Emotion Types**

Based on such emotion values calculated by EGC method and their situations, the degree of pleasure/displeasure for each emotion type is obtained. In this paper, we consider only 20 emotion types, among 24 emotion types defined by Elliot in [11], that is, "joy" and "distress" as a group of "Well-Being"; "happy-for," "grating," "resentment," and "sorry-for" as a group of "Fortunes-of-Others"; "hope" and "fear" as a group of "Prospect-based"; "satisfaction," "relief," "fears-confirmed," and "disappointment" as a group of "Confirmation"; "pride," "admiration," "shame," and "disliking" as a group of "Attribution"; "liking" and "disliking" as a group of "Attraction"; "gratitude," "anger," "gratification," and "remorse" as a group of "Attraction/Attribution"; "love" and "hate" as a group of "Attraction/Attribution." The emotion type "liking" and "disliking" are not included in the generated emotion type by EGC. Figure 6 shows the dependency between the groups of emotions type.

**V. EXPERIMENTATION**

For the test data, 38 university students (1st grade: 11, 2nd grade: 10, 3rd grade: 9, 4th grade: 8 / male: 27, female: 11) described their present annoyances freely. We apply our method into 34 annoying descriptions (except two examples because they do not contain any annoyances and two more examples because they are difficult to be classified into one annoying pattern even human). However, some input data are too short to apply our method because such data imply only a little words and generated multi-dimensional vector is almost 0. So our method cannot extract any features from such vectors. Therefore, we input such data after complimenting some words following the subject's intention when the input text is shorter than 60 characters. We experimented with the test data using the map which size was 20*20 and number of training epochs was 5,000,000. Table 4 shows the experimental result. Four test data were calculated their annoying patterns and three of them
were correct.

As shown in Table 4, most of test data could not be calculated their annoying patterns because the output were “unlabeled.” Then, we estimate the labels of such “unlabeled” nodes based on the relationship between the node and the surrounding nodes. We show an example of label estimation in Fig. 7. At first, we select a node which has the strongest relationship with target node from the surrounding nodes. The intensity of the relationship is displayed by the gradation of the relation hex. Then, the label of the target node is defined as the same of the selected node’s label.

![Fig.7 Relationship intensity between nodes](image)

Table 5 indicates the result of the experimental result using the label-estimated map. The labels of 33 test data were calculated and 16 of them were correct.

Finally, Figure 8 shows the results the contents and its degree by EGC. The bottom in this figure represents positive, negative, positive in the field of “Kurashi (living),” “Manabi (study),” and “Sigoto (job),” respectively.

<table>
<thead>
<tr>
<th>Answer</th>
<th>living</th>
<th>study</th>
<th>job</th>
<th>unknown</th>
<th>unlabeled</th>
</tr>
</thead>
<tbody>
<tr>
<td>living</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>study</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>job</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>30</td>
<td>34</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Answer</th>
<th>Living</th>
<th>Study</th>
<th>job</th>
<th>unknown</th>
</tr>
</thead>
<tbody>
<tr>
<td>living</td>
<td>2</td>
<td>10</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>study</td>
<td>1</td>
<td>13</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>job</td>
<td>0</td>
<td>4</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>27</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

TABLE IV

VI. Conclusion

In this paper, we proposed a method to calculate annoying pattern (living, study, and job) of the text of counseling content using the SOM trained by 90 training data. The training data contain 30 text data from each three annoying patterns and the data was collected from the description in the web based bulletin boards. We also proposed a method to generate 831-dimensional vector from input text data by analyzing morphologically and counting the number of feature words which featured the three annoying patterns. The SOM trained by 90 data was labeled based on the labels of the training data. When a unit has all the same type of annoying pattern labels or one type of label occupies the unit more than 50%, we give the unit the major label. When some type of labels share the node, the node has “unknown” label.

We experimented with our proposed method by using the trained and labeled map and the correct answers were obtained for three test data. Therefore, we prepared another map which the labels of “unlabeled” nodes were estimated and experimented with the map. Then, 16 correct answers were calculated. The developed system can show the results of type of their annoyances and the emotional degrees by EGC.

However, 16 examples are classified wrongly in the experiment using label-estimated map. One of the reasons is the lack of feature words. For the future work, we will develop our method to be able to add new feature words automatically from the failed test data.

REFERENCES

[1] mixi (http://www.mixi.jp/)