Eurasia Journal of Mathematics, Science & Technology Education, 2016, **12**(5), 1139-1151
doi: 10.12973/eurasia.2016.1502a

An Intelligent Recommendation System for Animation Scriptwriters' Education

Shang-Te Tsai
National Taiwan University of Science and Technology, TAIWAN
Ting-Cheng Chang
Ningde Normal University, CHINA
Yu-Feng Huang
Xiamen University, CHINA

•*Received 1 April 2015*•*Revised 1 June 2015* •*Accepted 10 June 2015*

Producing an animation requires extensive labor, time, and money. Experienced directors and screenwriters are required to design scenes using standard props and actors in position. This study structurally analyzes the script and defines scenes, characters, positions, dialogue, etc., according to their dramatic attributes. These are entered into a model developed by this research, called *animated behavior by understanding the script module* (ABUSM), which implements a design for scenes, props, and role-related databases, producing a corresponding picture. Then, through mutual correspondence between attributes, data mining theories calculate a module for each character and its interactions related to other characters. Finally, these modules are rendered into animation. This demonstrates the relationships between behaviors through an animation demonstration system. Our research results will help nonprofessional theater animation designers, providing a reference for designing and modifying scripts through animated presentation until they are satisfactorily developed. This reduces the cost, time, and labor in recording a drama or producing an animation and can serve as a reference or training for nonprofessional screenwriters and script designers.

Keywords: animation movement behavior, creative drama education, drama writing, script analysis, script writing

Correspondence: Yu-Feng Huang,
School of Journalism and Communication, Xiamen University, No. 422, Siming South Road, Xiamen, Fujian, China.
E-mail: mkk8653@my.fsu.edu

ISSN: 1305-8223 http://iserjournals.com/journals/ejmste

INTRODUCTION

Animation script creation

Script creation can be a form of literary creation, a textual foundation for theatrical art. The director and actors choreograph and perform according to descriptions in the script, acting out the text provided in the script, which is then viewed on a screen. The most important aspect of script creation is providing the means for an actor to demonstrate through performance. The performing art is not considered complete when the script is finished; rather, the ultimate presentation occurs through the performance of the screenplay. A script written by a playwright is the heart and soul of an animated film or TV series since it involves the story, screenplay, props, and dialogue. The script of a play represents not only the theatrical performance but also the significance of the story structure and the artistic human spirit. Screenplays are generally classified as drama, film, television, animation, game play, etc., and based on the subject of the script, they can be subdivided into comedies, tragedies, historical dramas, family dramas, thrillers, and so on.

Generally speaking, when creating a script for an animation, the creator should have a complete basic story in mind, including the origin, development, and ending, as well as the roles and structure. This basic structure will be rendered into a summary, which shows a clear plot by explaining story development and providing many details. Then, ideas are classified to develop storylines and create scenes, which comprise story segments; these story segments are then organized in a linear manner with three parts: beginning, middle, and ending. These segments are gradually put into the linear structure until all story segments are concatenated into a complete linear organization. The story segments can be said to comprise the scenes of the script, and in each scene the story needs to be presented with the relevant contextual elements, such as themes, settings, characters, props, dialogue, narration, time, and so forth. An experienced playwright can usually arrange dramatic elements in a competent way so the story has the appropriate rhythm and dramatic climaxes.

State of the literature

- This description is more of an abstract, rather than intuitive, way of showing such aspects of a role.
- Shooting and producing an animation is usually quite labor intensive, requiring a lot of time and money. An experienced director and screenwriter are needed to design a scene with standard props and corresponding actors in position, ready to perform dramatic behaviors such as emotional expression and dialogue.
- The present study analyzes the script in a structured way and defines scenes, characters, positions, dialogue, etc., according to their dramatic attributes. These attributes are then entered into a model, developed by this research, called animated behavior by understanding the script module (ABUSM), which implements a design for scenes, props, and role-related databases that result in a corresponding picture.

Contribution of this paper to the literature

- It is hoped that the results of this research can be helpful for nonprofessional theater animation designers.
- This study provides a reference for designing and modifying a script through animated presentation until it is satisfactorily developed.
- This can reduce the cost, time, and labor required to shoot a drama or produce an animation, and can serve as a reference for directors and screenwriters, or for the training of nonprofessional screenwriters and script designers.

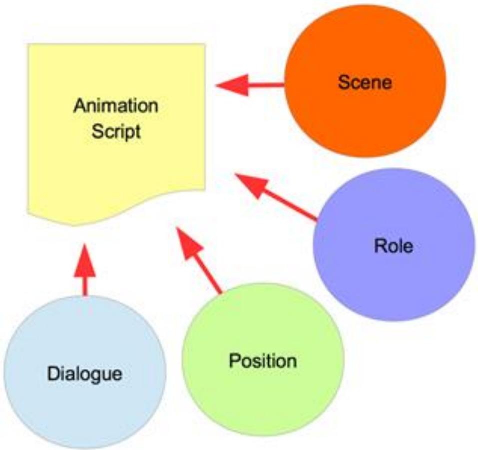


Figure 1a. Constituent elements of a script

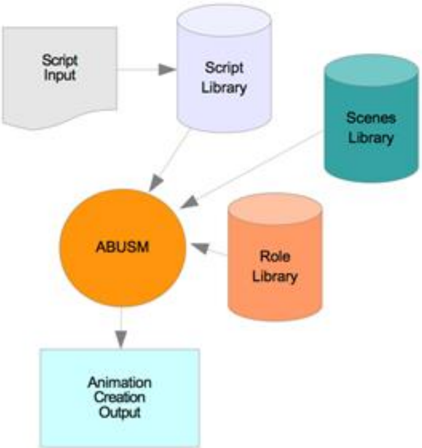


Figure 1b. Animation scripts and database design elements generating the animation architecture

Script structure analysis

This research found that, in general, when a playwright creates a script, story elements are presented in a structured way to clearly express the storyline, and the structure of these elements has a specific format. Playwrights use a basic framework when creating a script, which at the very least contains settings, characters, positions, and dialogue (Marx, 2007). These constitute the basic dramatic elements (Figure 1a) used to construct the story the playwright has in mind. Playwrights use these elements to create their scripts and screenplay structures; such formatting makes it easier for directors and actors to read and understand the script (Davies, 2008), thus facilitating a smooth production.

Screenplay analysis for generating animation

This research analyzed the content of scripts to discover the elements that comprise scenes and use those elements to identify the interactions between attributes. These properties are then transferred into a related systems database to make calculations and produce scenes, characters, positions, dialogue, and other animated design elements (Van Wie et al., 2007). Finally, these elements are combined to create animated images that can be experienced visually.

DATABASE SYSTEM STRUCTURE FOR ANIMATION SCRIPTS AND DESIGN ELEMENTS

This research designed scripts, settings, characters, and other databases using a cloud environment. This can be very convenient for uploading design works to corresponding cloud databases, using the characteristics of computing structure (Goldner & Birch, 2012). This allows any user with a computer connected to the Internet anywhere in the world to easily preserve his or her own database creations and share resources, thus creating a synergistic creative environment (Anonymous Pixton Comics Inc., 2014).

Scripts written by playwrights are uploaded to a script database to generate the desired animation elements—such as setting, character, and design (Pierson & Rodger, 1998)—and provide creators with a database for uploading settings and character designs (Li, Liao, & Liao, 2004). The design elements collected from these creators are integrated into a cloud-based system to facilitate the full utilization of these resources (Maiocchi & Pernici, 1990). The *animated behavior by understanding the script module* (ABUSM) is used to identify the constituent elements and attributes of the scripts uploaded to the cloud system. Setting and character databases are linked and matched in a process similar to a director searching for actors and matching them with scenes. After the match, the corresponding design

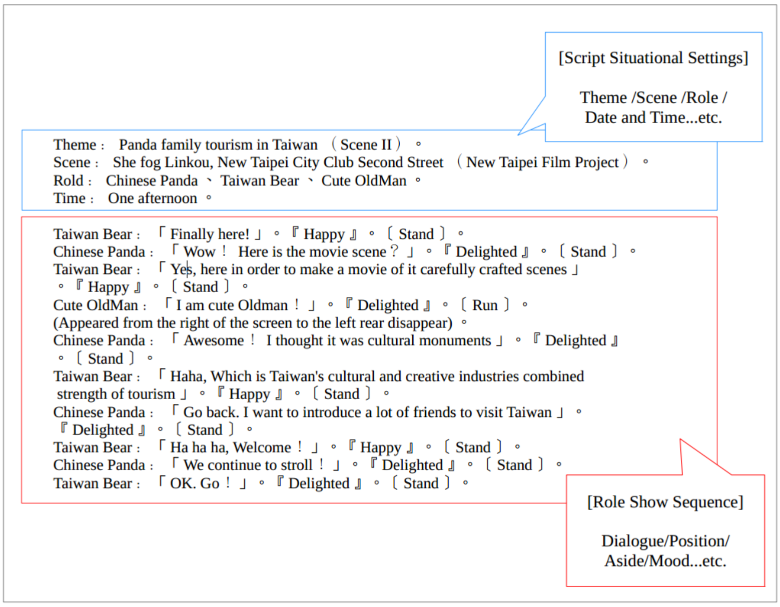


Figure 2a. Chinese script format structure

Table 1. Script control symbols

Symbol	Explanation
◆	Aside
「」	Dialogue
：	Role select
『』	Facial expression
〔〕	Action
()	Position
。	Paragraph

elements produce corresponding styles. The behavioral pattern for each character's position and dialogue is calculated according to the descriptions to obtain the interactions between characters. The corresponding role modules are then applied to complete the dialogue and interactions among characters. Finally, design elements such as settings, characters, and scenes are combined to produce an animation (Figure 1b).

ANALYZING SCRIPT-GENERATED ANIMATION

Analyzing the format structure of Chinese scripts

This study used Chinese scripts to analyze the structure and format of scripts. The research found that the average screenplay format is divided into two major components to determine the direction of theatrical performances. The first is the top-of-script format, which includes theatrical themes, scenes, characters, date and time, etc.; the second format is found below the top part and includes dialogue, positions, narration, emotion, etc. (Figure 2a). The lower part also includes symbols used to indicate dramatic action (Table 1).

The Chinese Word DNA Dictionary

The Chinese Word DNA Dictionary (Bong-Foo, 2014) pertains to a theory advanced by Chu Bong-Foo (2015) that the structure of Chinese characters expresses the cognition of ancient Chinese people. Characters can be sorted and grouped into conceptual and common-sense categories (Figure 2b) to reveal their meaning. This theory can be applied to the structure of computer information systems such that Chinese characters can be used to effectively manage and control such systems. Chu Bong-Foo suggested that Chinese characters have six elements and functions. These include shape, pronunciation, and meaning, as well as the following developed in response to the computer information age: word codes, word order, and word differentiation. Analyzing Chinese characters to identify their genes is referred to as Chinese Word DNA Engineering (Table 2).

REVIEW OF BASIC METHODOLOGIES

Rough sets

Pawlak (1982) first introduced rough set theory. Rough set theory is a powerful mathematical tool for handling the vagueness and uncertainty inherent in making decisions. The theory is founded on the assumption that every object in the world of discourse is associated with some information. Objects characterized by the same information are indiscernible (similar) in view of the available information.

(1) Information System

In rough set theory, information systems are used to represent knowledge. An information system $S = (U, \Omega, V_q, f_q)$ consists of:

U : a nonempty, finite domain set

Ω : a nonempty, finite set of attributes

Ω = $C \bigcup D$, in which C is a finite set of condition attributes, and D is a finite set of decision-making attributes. For each $q \in R$, V_q there is a definition domain of q ; f_q is the information function ; $f : U \rightarrow V_q$.

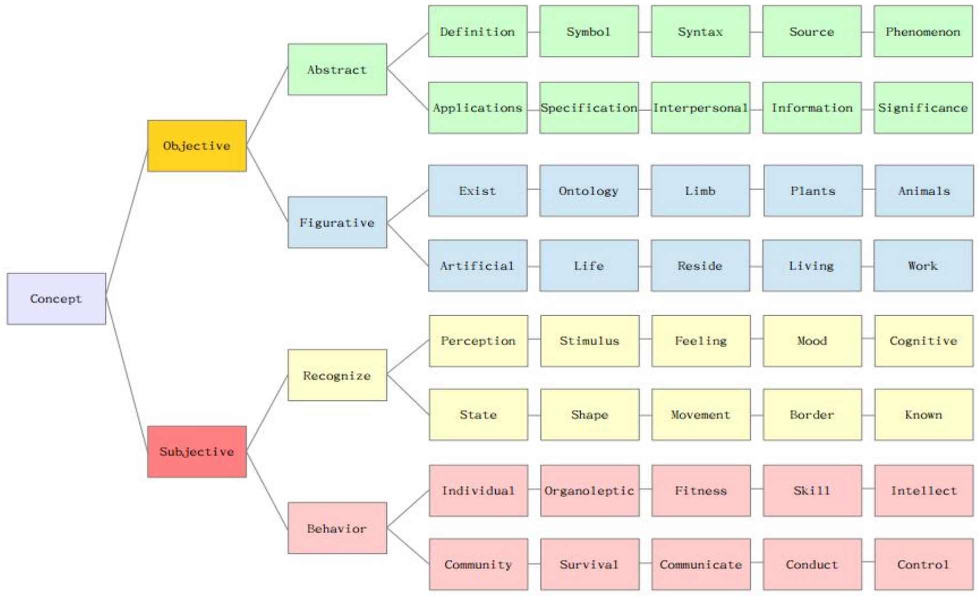


Figure 2b. Chinese script format structure

Table 2. Six elements of Chinese words

Factor	Function
Word Code	Word Encoding
Word Order	Searches for Word Sort
Word Font Shape	Visual Perception of Chinese Word
Word Resolution	Word Shape and Meaning of Abstraction in Dynamic Process
Word Pronunciation	Pronunciation of the Chinese Word and Feelings it Produces
Word Meaning	Chinese Word for Human Cognition

(2) Approximation of Sets

There are often some inexact, unclear parts in real-world data. Therefore, the inconsistencies among elements (X) appear to generate ($IND(R)$). Thus, it is hard to discern the relations or differences among two or more elements with various decision-making attributes in the condition set. The decision-making table is henceforth called the indiscernibility table. However, the major function of approximate sets in rough set theory is to deal with this indiscernibility of elements. This decision table is called an inconsistent decision table. In rough set theory, approximations of sets are introduced to deal with inconsistency. If $S=(U,\Omega,V_q,f_q)$ is a decision table, $R\subseteq\Omega$ and $X\subseteq U$, then R^- and R_- are the upper and lower approximations of X and are defined, respectively, as follows:

$$R^-(X)=\bigcup\{Y\in U/IND(R):Y\cap X\neq\phi\} \quad (1)$$

$$R_-(X)=\bigcup\{Y\in U/IND(R):Y\subseteq X\} \quad (2)$$

Herein, $U/IND(R)$ expresses the equivalence of R. $IND(R)$ is the indiscernibility of R. They are defined as follows: $IND(R)=\{(x,y)\in U^2 : \text{for every } a\in R, a(x)=a(y)\}$ (3)

When using the attribute set R, the lower approximate set R_-X describes the set of the completely same-ranked elements (X) under the Y decision-making

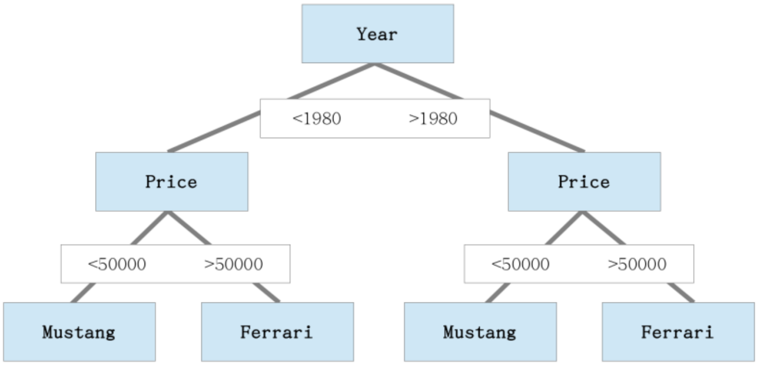


Figure 3a. Decision tree for the problem *choosing a car*

attribute; $R^{-}X$ represents the set of possible same-rank elements (X) under the Y decision-making attribute. The set $BN_R(X) = R^{-}(X) - R_{-}(X)$ is called the boundary set of X .

Decision tree

In machine learning, knowledge is extracted from a training sample for future prediction (Borgonovo & Tonoli, 2014). Most machine learning methods make accurate predictions but are not interpretable. In this study, we concentrated on decision trees, which are simple and easily comprehended. They are robust to noisy data and can learn disjunctive expressions (Mitchell, 1996). The entire set of examples is split into subsets that are easier to handle. The type of the split determines type of the decision tree. The split can be based on one feature (Quinlan, 1986) or a linear combination of features (Breiman et al., 1984).

An example of data (training set) for the problem *choosing a car* is shown in Figure 3a.

(1) Algorithm for tree construction

The basic tree construction algorithm is a greedy search through the space of possible decision trees. The search starts by creating a root node and continues by processing the training set according to the following steps:

1. If all instances belong to the same class C_i, stop and return the leaf node with class C_i.
2. Find the split that best classifies the instances (as will be explained later).
3. Each split divides the data into subsets. (For example, the split *year = 1980* divides the data into two subsets with seven and nine instances in each.)
4. For each subset of data, repeat steps 1, 2, 3, and 4 to construct the decision tree.

Thus, this algorithm is recursive. For each node of the tree, it is called once. Finding the best split takes the most time. The time complexity of the algorithm is O (*l* * time complexity [best split function]), where *l* is the number of nodes in the decision tree.

(2) Identification trees (ID3)

Identification trees classify instances by sorting them down the tree from the root to some leaf node. Each internal node in the tree specifies a test of *one* attribute of the instance, and each branch descending from that node corresponds to one of the possible values or intervals for that attribute. An instance is classified by starting at the root node of the tree, testing the attribute specified by that node, and then moving down the corresponding tree branch. This process is recursively repeated for the subtree of the new node until a leaf node is reached. The leaf node stores the class code.

In the ID3 algorithm (Brodley & Utgoff, 1995), the best split is found as follows:

1. For each attribute x_i, do the following:

- If the feature is symbolic with m possible values, the instances are divided into m groups, wherein each group the instances have the same value for the attribute x_i. Calculate the partition-merit criteria as p_i.
- If the feature is numeric, the instances can be divided into two in k different ways, where k is the number of different values of the attribute x_i. For each of these k ways, the partition-merit criterion is computed and the best is selected as p_i.

2. Find the attribute j such that $p_j = min_i \, p_i$ is the split node attribute.

- If x_i is symbolic with m possible values, partition the set of instances into m subsets where at each partition $x_j = a_k$, $k = 1, \ldots, m$.
- If x_i is numeric, partition the set of instances into two: $x_j \leq a$ and $x_j > a$, where a is the split threshold that optimizes the partition-merit criterion.

This algorithm has a complexity of O ($f * n$), where f is the number of attributes and n is the number of instances.

To classify the instances in the test set, we start from the root node and trace the tree node by node. At each internal node, we take the subtree for which the instance has the correct attribute value or is on the correct side of the split threshold. At each leaf node, if the instance has the same class as the class of the leaf node, the instance is correctly classified; otherwise, it is misclassified.

ANALYSIS PROCESS FOR ROLE ACTION IN ANIMATION SCRIPTS

This research entered Chinese Word DNA Dictionary data into a database that collected story information for about 4,000 Chinese animation scripts. The database was analyzed using the knowledge data set. Preliminary analysis produced information system tables for the properties; then, the lower approximations of a set calculated the BNX. Stories that did not conflict were calculated through the decision tree to predict the behavior of a role. Finally, it was checked whether the role actions matched the results calculated by the system. If the results matched, the character's actions would be produced; if not, these data were discarded or returned to the system database to supply other story calculations.

Based on the theories mentioned above, we developed a standard process to analyze character emotions, dialogue, and positions, and then recommend dramatic situations that might occur to the playwright as a reference for modifying the script. Below, we explain data processing step by step from beginning to end. Details are shown in Figure 3b as follows:

(a) Flowchart of model establishment (Figure 3b)

(b) Steps in the modeling flow

Step1.Knowledge database: We established a database of more than 4,000 animation scripts.

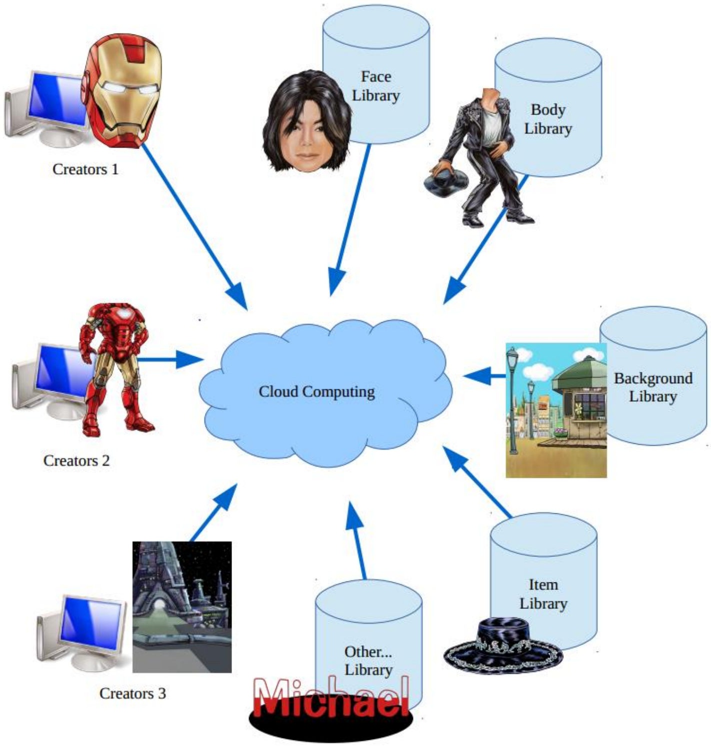
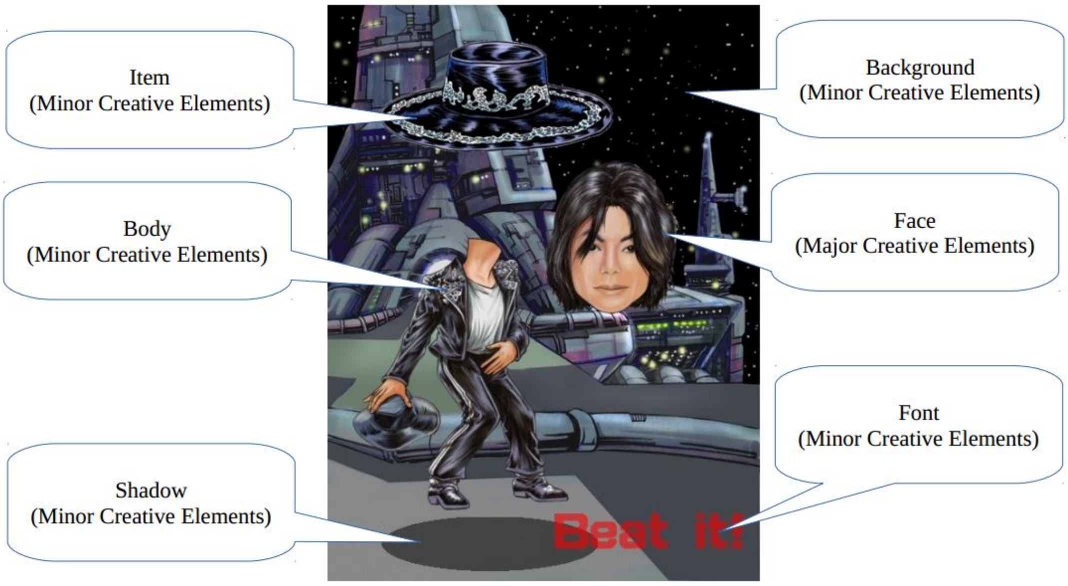


Figure 4a. Design architecture of cloud-based shared authoring system

Figure 3b. Analysis process for role action in animation scripts



Explanation
Aside
Dialogue
Role Select
acial Expression
Action
Position
Paragraph

Figure 4b. Analysis of creative elements

Step 2. Information System Table: First, we needed to set up objects to be explored and make decisions about conditional attributes and decision-making attributes. Since the selection of conditional attributes focused on evaluating the quality of a script, we suggested using four classes to conduct selection: asides, dialogue, role selection, and facial expressions. The decision-making attributes were the role responses in the script, such as action, position, and paragraph period. Each attribute had a different number of clusters; the clustering results are shown in Table 3.

Step 3. Lower Approximation Set ($\underline{RX}$) of Rough Set: We obtained the lower approximation set through the rough set. A collection of 80%

of the elements in the lower approximation set was used as training data for the decision tree (DT). The remaining 20% were taken as testing data.
Step 4. Three Decision Tree Classifiers:
Three four-input/one-output decision tree classifiers were used as the learning machine to forecast the next movement in the animation. All inputs were the same for the three classifiers. The output was either an action, a position, or the end of a paragraph.

Step 5. Checking the Results: The proposed process can be implemented provided the test results are accepted. Otherwise, you go back to Step 1 and revise it.After these steps are completed, two decisions must be made. First, it should be checked whether the proposed model is applicable to animation. If not, renew the attributes determination. Second, the subsequent stage should be created based on the next movement.

SYSTEM IMPLEMENTATION AND DEMONSTRATION EXAMPLES

Design architecture of the cloud-based shared authoring system

This research utilized a cloud system. An "online modeling database" put role designs into categories such as face, body, background, and item, as well as other modeling styles such as the shadows or fonts. Creators could upload their work within these categories according to their own preferences and modeling styles, and exchange designs (Ali, Razool, & Bloodsworth, 2012). This system integrated these designs (Lu, Wen, & Sun, 2012), creating a coauthored (Etter, 1991) "online modeling database" for the creators' convenience (Figure 4a).

Analysis of creative elements

This study proposes that a complete model of comic roles, including major roles and backgrounds, can be composed of several "design elements" in design models; these design elements are composed of "major design elements" and "minor design elements." The "major design elements" are the main elements constituting the shape of a comic the soul of the whole design. In accordance with the "main design elements," other "minor design elements" contribute toward producing a complete model. Taking Michael Jackson's face as the main design element, for example, one hopes the system will find minor design elements related to how Michael Jackson dances (Figure 4b). Once the system finds the appropriate design elements, it integrates the shapes and produces a complete cartoon of Michael Jackson dancing (Figure 5a).

Online design elements in a shared database system

This research was designed to develop online scenes and character databases in the cloud community (Figure 5b). The system allows creators to easily upload their own design work and provides a communication platform between users (Zwass, 2010). As such, it achieves the purpose of sharing design elements between creators, thereby promoting collaborative creation and the exchange of resources (Shilba & Kiruba, 2013).

Editing the script and generating animation

Once the playwright enters a Chinese script, the system follows the elements of the script, links to the cloud-based design elements database, and produces an animation accordingly (Paracha et al., 2008). The image generated by the script is immediately available to the playwright on the system interface (Figure 6a). The system will follow the format of the script and story in its broadcast. If the

Figure 5a. The system generates the comic

Figure 5b. Online scene and role database system

Figure 6a. Script-generated animation

playwright modifies the script, the system immediately changes the animation and provides suggestions (e.g., modifying the content of a character's dialogue and the emotions involved). The system will display the role's facial expressions to provide a reference for the playwright and modify them accordingly (Figure 6b). If the playwright wants to modify the positions of the roles, the system will immediately display the new positions (Figure 7a). This research provides a visual environment for playwrights, helping them with the more abstract and imaginative aspects of script writing. It gives them processed animation playback for reference as well as timely suggestions for ways to edit script content, such as the expression or position of a role.

CONCLUSIONS AND FUTURE WORK

This research integrated cloud databases, the Chinese Word DNA Dictionary, and other systems to develop a screenplay-based input that can produce an *animated behavior by understanding the script module* (ABUSM). This system can animate motion pictures and recommend content based on the script, providing creators with a new kind of writing environment for professional or nonprofessional productions. Creators can refer to a demo screen produced in real time by the system, thereby reducing the cost, time, and manpower needed to shoot a drama or animation.

Since this system uses the Chinese Word DNA Dictionary, the current ABUSM is only capable of analyzing Chinese scripts. Therefore, the expectation for the future is that the module can be developed into a multilingual system. This would involve an

Figure 6b. System-recommended role moods

Figure 7a. System-recommended role positions

extended analysis of a national language, its grammar, and other related conditional functional modules. The result would be the provision of additional script languages that can be entered and rendered into animation. Then, the system could be implemented in drama or multimedia education, providing animation directors, playwrights, and others with access to such a system.

REFERENCES

Ali, Z., Rasool, R., & Bloodsworth, P. (2012). Social networking for sharing cloud resources. *Second International Conference on Cloud and Green Computing (CGC), 2012*, 160-166.

Anonymous Pixton Comics Inc. (2014). "Collaborative comic creation" in patent application approval process. *Telecommunications Weekly,* 184.

Bong-Foo, C. (2014). *Laboratory of Chu Bong-Foo*. Available at: http://www.cbflabs.com.

Borgonovo, E., & Tonoli, F. (2014). Decision-network polynomials and the sensitivity of decision-support models. *European Journal of Operational Research, 239*(2), 490.

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). *Classification and regression trees*. Belmont, CA: Wadsworth.

Brodley, C. E., & Utgoff, P. E. (1995). Multivariate decision trees. *Machine Learning, 19*, 45–77.

Chu Bong-Foo. (2015). In *Wikipedia*. Retrieved from http://en.wikipedia.org/wiki/Chu_Bong-Foo

Davies, E. (2008). The script as mediating artifact in professional theater production. *Archival Science, 8*(3), 181–198.

Etter, M. C. (1991). Hydrogen bonds as design elements in organic chemistry. *The Journal of Physical Chemistry, 95*(12), 4601–4610.

Goldner, M., & Birch, K. (2012). Resource sharing in a cloud computing age. *Interlending & Document Supply, 40*(1), 4–11.

Li, T. Y., Liao, M. Y., & Liao, C. F. (2004). An extensible scripting language for interactive animation in a speech-enabled virtual environment. *International Conference on Multimedia and Expo, 2004, 2*, 851–854.

Lu, Z., Wen, X., & Sun, Y. A. (2012). Game theory based resource sharing scheme in cloud computing environment. *World Congress on Information and Communication Technologies* (WICT), *2012*, 1097-1102.

Maiocchi, R., & Pernici, B. (1990). Directing an animated scene with autonomous actors. In *Computer Animation,* Jan. 1990, (pp. 41-60). Japan: Springer.

Marx, C. (2007). *Writing for animation, comics, and games*. Taylor & Francis.

Mitchell, T. (1996). *Machine Learning*. McGraw-Hill.

Paracha, S., Mohammad, M. H., Khan, M. T. A., Mehmood, A., & Yoshie, O. (2008). Balancing fabula & interactivity: An approach for narrative creation in interactive virtual drama. *4th International Conference on Emerging Technologies*, *2008*, 321–326.

Pawlak, Z. (1982). Rough sets. *International Journal of Information and Computer Sciences, 11*(5), 341–356.

Pierson, W. C., & Rodger, S. H. (1998). Web-based animation of data structures using JAWAA. *ACM SIGCSE Bulletin, 30*(1), 267–271.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning, 1*, 81–106.

Shilba, G. J., & Kiruba, S. A. (2013) Improving personalized image retrieval from the photo sharing social networks. *International Journal of Marketing and Technology, 3*(6), 32-45.

Van Wie, M., Stone, R. B., Thevenot, H., & Simpson, T. (2007). Examination of platform and differentiating elements in product family design. *Journal of Intelligent Manufacturing, 18*(1), 77.

Zwass, V. (2010). Co-creation: Toward a taxonomy and an integrated research perspective. *International Journal of Electronic Commerce, 15*(1), 11–48.

◈◇◈