Learning Text to Model: A Bayesian Network based L-system Modeling Strategy

Cheng Chen, Genlin Ji, Bin Zhao
School of Computer Science and Technology
Nanjing Normal University
Email: withchencheng@qq.com

***Abstract*—L-system is a prevailing modeling method for generating fractals, especially self-similar patterns such as plants. However it's too hard to design an appropriate L-system to get the desired visual models of plants. In order to generate a favorable plant model, usually we need to deduce backwards or guess the production rules of the L-system and then try to modify some control parameters over and over again. Inspired by information extraction technology, we propose a new strategy to model visual plants. We use Bayesian Networks to extract structured information describing the plant characters from user given text first, then we use that information to automatically generate an L-system alphabet, axiom and production rules. Comprehensive experimental evaluation conducted on real botanic text corpora demonstrates that our proposal is very helpful in artistic plants modelling.**

I. Introduction

L-systems were introduced and developed in 1968 by Aristid Lindenmayer to describe the behavior of plant cells and to model the growth processes of plant development [1]. An L-system or Lindenmayer system is a parallel rewriting system and a type of formal grammar. The recursive nature of the L-system rules leads to self-similarity and thereby, fractal-like forms like plant models are easy to describe with an L-system [2]. However it's non-intuitive to determine the details of the L-system to generate a desired plant model. In fact, one may try to modify the rules many times to get the right L-system axiom and productions. That is tedious and boring when you only have some immature thoughts about what you want to create. Thus we propose a new strategy using Information extraction (IE) to construct L-system for visual plants generating. Information extraction distills structured data which charaterize the plants model from unstructured user input text by identifying botanic entities as well as their stated properties. Because the information extraction task here is domain specific, experiments show that even hand-written patterns is highly effective in extracting the plant characters, let alone the Bayesian Networks. Then the use of L-system is proposed, with some of its extensions representing the biological features [3], to simulate the virtual plants model.

II. Information Extraction and L-system Generating

An L-system consists of an alphabet of symbols that can be used to make strings, a collection of production rules that expand each symbol into some larger string of symbols, an initial axiom string from which to begin construction, and a mechanism for translating the generated strings into geometric structures. An L-system is defined as a tuple

$$G = (V, \omega, P)$$

where 1) V (the alphabet) is a set of symbols containing both variables and terminals; 2) ω (axiom) is a string of symbols from V defining the initial state of the system; 3) P is a set of production rules defining the way variables can be replaced with combinations of terminals and other variables. If any character c in the current string mathches the predecessor of any production, the c will be replaced with the corresponding successor. Using L-systems for generating graphical plant images requires that the symbols in V refer to elements of a plant on the computer screen, for example, a green leaf, a red flower, or a brown, thin branch pointing to the sky.

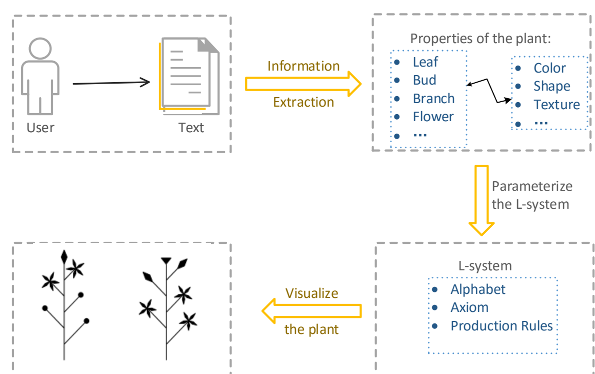


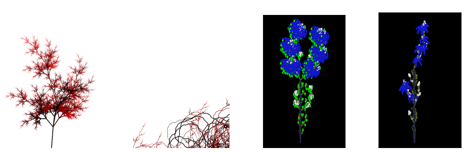
Fig. 1. Workflow of our approach

One of the important issues is property extraction of the plants from user given text. Hand-written regular expressions will suffice to do this job most of the time. However, due to the diversity of contexts where the desired properties can appear, it's tedious and difficult to manually develop patterns for extraction in a robust system. Bayesian network is a more advanced method, and it outperforms the former naive regular expressions. A BN is a graphical representation of probability distributions. A lot of work has been done for learning both the structure and the parameters of a BN from a data set. The extraction module takes domain ontology and unstructured text as input and performs extraction using rules by exploiting knowledge stored in ontology. The extracted data generally contains some missing values. For example, some plants just don't fruit, and some corpora just miss the size of the fruit. The integration of BN aims to address this issue. Furthermore, the integration can aid in resolving different types of conflicts present in the extracted data. We accomplish the learning of BN using BN-PowerConstructor[1]. The tool learns the structure and parameters of a BN using heuristics.

After semantic processing of the text, we run the Bayesian network based on ontology annotation. Fig. 1 shows the workflow of out approach, which consists of the following three main steps.

1) Extract information describing properties of the plant from user given text. We do pre-annotation, semantic annotation, text annotation first, and then apply Bayesian network to the corpora.
2) Use that information to determine the axiom, production rules, and assign appropriate action meanings for each terminal in the alphabet based on the properties of the plant.
3) Run the L-system in step 2 to generate the visual plant model.

Fig. 2 shows some demo results of our text to plant model system.



(a) maple (b) vine (c) flower (d) grass

Fig. 2. Demonstrations of our text to plant model system

The production rules for a complex and beautiful plant as subfigure (c) is amazingly simple:

$$FF - [-F + F + FB] + [+F - F - F]$$

Where 'F' means drawing forward; 'B' means drawing a bud; '$+$' means turn left some degrees('$-$' for right). The square bracket ' [' commands the system to save the current state, which are restored when the corresponding '] ' is executed. The colors, shapes, and angles are generated automatically from user given text.

III. Experiment Results and Discussions

Many datasets describing plants are available on the internet. For our work, we choose some specific plants entries in Wikipedia such as maple, wisteria sinensis, and epipremnum aureum. The morphological description part is our unstructured data source. we randomly divided the whole dataset into two parts by 70% (training set) and 30% (test set), using balanced F_1 measure to evaluate the performance of our text to plants model system. And the score of a test result is given by

$$F = 2PR/(P + R) + B$$

where F is the performance score; P (precision) means the fraction of represented characteristics that are correct; R (recall) means the fraction of correct characteristics that are represented; and A stands for the aesthetics of the output plants model.

We classify the plants sketchily into three categories: trees, vines, and grass. More detailed characters include branch, flower, leaf, internode, bud, etc, denoted by $d_0, d_1, \ldots, d_n$. And the P, R scores for one category are the mean values of d_i. Each d_i has various judging metrics such as branch angle, flower color, leaf texture, denoted by $m_0, m_1, \ldots$. The overall score for d_i is calculated as the mean value of m_i. Table I shows the experimental results of our approach. The IE method used here is Bayesian Networks.

TABLE I
Measurement of our approach

Details	branch	flower	leaf	internode	F_1 value
Tree	0.81	0.90	0.84	0.77	0.8125
Vine	0.79	0.61	0.93	0.78	0.7775
Grass	0.72	0.87	0.89	0.91	0.8475

By far, we have designed a three step model to construct L-systems for visual plants generating using Bayesian Networks from user given textual description of the plant. However, it's two-dimensional and static in space and time. In [5], they use an evolutionary algorithm to evolve interpreters for Lsystems. We can further extend our model to simulate the dynamics of plants from a given status. It becomes more interesting when the input is a scene consisting amounts of plants, under which circumstance we need to determine not only the characters of every single plant but also their reciprocal actions and the surrounding environment. In the future, we'll study the translation from text to plants dynamics, evaluate our approach on other datasets, as well as develop new L-system programming languages.

References

[1] A. Lindenmayer. "Mathematical models for cellular interaction in development, Parts I and II". *Journal of Theoretical Biology*, 18:280-315, 1968.

[2] Erick Castellanos and Félix Ramos and Marco A. Ramos, "Semantic death in plant's simulation using Lindenmayer systems", *International Conference on Natural Computation* , 360–365, 2014.

[3] Przemyslaw Prusinkiewicz, Mark Hammel , Jim Hanan, and Radomír Měch, " L-Systems: From the Theory to Visual Models of Plants", *2nd CSIRO Symp osium on Computational Challanges in Life Sciences*, 1996.

[4] R. Feldman, and I. Dagan, "Knowledge discovery in textual database(KDT)", *The first international conference on knowledge discovery and data mining (KDD) Montreal, Canada AAAI press, pp*, 112-117, August 1995.

[5] Daniel Ashlock, Kenneth M. Bryden, "Evolutionary control of Lsystem interpretation". *IEEE Congress on Evolutionary Computation*, 2004: 2273-2279.

[1] The details of BN-PowerConstructor can be found on http://web.engr.oregonstate.edu/∼dambrobr/uai-archive-pre00/0258.html