2011 International Conference on Information Communication and Management
IPCSIT vol.16 (2011) © (2011) IACSIT Press, Singapore

Text Summarization Based on Cellular Automata

Farshad Kiyoumarsi[1], Fariba Rahimi Esfahani[2], Pooya Khosraviyan Dehkordi[3]

[1]Islamic Azad University-Shahrekord Branch, Shahrekord,Iran
Kumarci-farshad@iaushk.ac.ir
[2]Islamic Azad University-Shahrekord Branch, Shahrekord,Iran
Rahimi_f@iaushk.ac.ir
[3]Islamic Azad University-Shahrekord Branch, Shahrekord,Iran
Khosravyan@iaushk.ac.ir

Abstract. This work proposes an approach to address the problem of improving content selection in automatic text summarization by using some statistical tools. This approach is a trainable summarizer, which takes into account several features, for each sentence to generate summaries. First, we investigate the effect of each sentence feature on the summarization task. Then we use all features in combination to train Cellular Automata (CA), genetic programming approach and fuzzy approach in order to construct a text summarizer for each model. Furthermore, we use trained models to test summarization performance. The proposed approach performance is measured at several compression rates on a data corpus composed of 17 English scientific articles.

Keyword: Text summarization, Cellular Automata, Machine Learning.

1. Introduction

Automatic text summarization has been an active research area for many years. Evaluation of summarization is a quite hard problem. Often, a lot of manual labour is required, for instance by having humans read generated summaries and grading the quality of the summaries with regards to different aspects such as information content and text clarity. Manual labour is time consuming and expensive. Summarization is also subjective. The conception of what constitutes a good summary varies a lot between individuals, and of course also depending on the purpose of the summary.

Recently many experiments have been conducted for the text summarization task. Some were about evaluation of summarization using relevance prediction [6], and voted regression model [5]. Others were about single- and multiple-sentence compression using ‘‘parse and trim” approach and a statistical noisy-channel approach [18] and conditional random fields [12]. Other research includes multi-document summarization [4] and summarization for specific domains [10].

We employ an evolutionary algorithm, Cellular Automata (CA) [11], as the learning mechanism in our Adaptive Text Summarization (ATS) system to learn sentence ranking functions. Even though our system generates extractive summaries, the sentence ranking function in use differentiates ours from that of [15] who specified it to be a linear function of sentence features. We used CA to generate a sentence ranking function from the training data and applied it to the test data, which also differs from [8] who used decision tree, [1] who used Bayes’s rule, and [12] who implemented both Naïve Bayes and decision tree.

In this work, sentences of each document are modeled as genetic programming of features extracted from the text. The summarization task can be seen as a two-class classification problem, where a sentence is labeled as ‘‘correct” if it belongs to the extractive reference summary, or as ‘‘incorrect” otherwise. We may give the ‘‘correct” class a value ‘1’ and the ‘‘incorrect” class a value ‘0’. In testing mode, each sentence is given a value between ‘0’ and ‘1’ (values between 0 and 1 are continuous). Therefore, we can extract the appropriate number of sentences according to the compression rate. The trainable summarizer is expected to ‘‘learn” the patterns which lead to the summaries, by identifying relevant feature values which are most correlated with the classes ‘‘correct” or ‘‘incorrect”. When a new document is given to the system, the ‘‘learned” patterns are used to classify each sentence of that document into either a ‘‘correct” or ‘‘incorrect” sentence by giving it a certain score value between ‘0’ and ‘1’. A set of highest score sentences are chronologically specified as a document summary based on the compression rate.

2. Background

2.1. Text feature

We concentrate our presentation in two main points: (1) the set of employed features; and (2) the framework defined for the trainable summarizer, including the employed classifiers.

A large variety of features can be found in the text-summarization literature. In our proposal we employ the following set of features [3,9]:

(F1) Sentence Length. (F2) Sentence Position. (F3) Similarity to Title. (F4) Similarity to Keywords. (F5) Occurrence of proper nouns. (F6) Indicator of main concepts. (F7) Occurrence of non-essential information. (F8) Sentence-to-Centroid Cohesion.

2.2. Text summarization based on genetic programing

In order to implement text summarization based on Genetic Programming [2], we used GP since it is possible to simulate genetic programming in this software. To do so; first, we consider each characteristic of a text such as sentence length, location in paragraph, similarity to key word and etc, which was mentioned in the previous part, as the genes of GP. Then, we enter all the operators needed for summarization, in the knowledge base of this system (All those operators are formulated by several expends in this field). After ward, a value from zero to one is obtained for each sentence in the output based on sentence characteristics and the available operators in the knowledge base. The obtained value in the output determines the degree of the importance of the sentence in the final summary. To do these steps, we summarize the same text using genetic programming.

2.3. Text summarization based on fuzzy logic approach

In order to implement text summarization based on fuzzy logic [7], we used MATLAB since it is possible to simulate fuzzy logic in this software. To do so; first, we consider each characteristic of a text such as sentence length, location in paragraph, similarity to key word and etc, which was mentioned in the previous part, as the input of fuzzy system. Then, we enter all the rules needed for summarization, in the knowledge base of this system (All those rules are formulated by several expends in this field). After ward, a value from zero to one is obtained for each sentence in the output based on sentence characteristics and the available rules in the knowledge base. The obtained value in the output determines the degree of the importance of the sentence in the final summary. To do these steps, we summarize the same text using fuzzy logic.

2.4. Cellular automata

At the beginning of 1950, cellular automata (CA) have been proposed by Von Neumann. He was interested to male relation between new computational device - automata theory -and biology. His mind was preoccupied with generating property in natural events [13].

He proved that CA can be general. According to his findings, CA is a collection of cells with reversible states and ability of computation for everything. Although Van rules were complicated and didn’t strictly satisfy computer program, but he continues his research in two parts: for decentralizing machine which is designed for simulation of desirable function and designing of a machine which is made by simulation of complicated function by CA [13].

Wolfram has conducted some research on problem modeling by the simplest and most practicable method of CA architecture too. In 1970,"The Game of Life" introduced by Conway and became very widely known soon. At the beginning of 1980, Wolfram studied one-dimension CA rules and demonstrated that these simple CAs can be used in modeling of complicated behaviors [16].

2.4.1. Definition

CA is characterized by (a) cellular space (b) transfer rule [11]. For CA , cell, the state of cell in time t, sum of neighbors state at time t and neighborhood radius are denoted by i, S_i^t ,η_i^t , and r, respectively. Also, the rule is function of $\varphi(\eta_i^t)$.

2.4.2. Change state rules

Each cell changes its state, spontaneously. The primary quality of cells depends on primary situation of problem. By these primary situations, CA is a system which has certain behavior by local rules. The cells which are not neighbors, have no effect on each other. CA has no memory, so present state defines the next state [16].

Quad rule CA is as CA= (Q, d, V and F), where Q, d, V and F are collection of possible state, CA dimension, CA neighborhood structure and local transferring rule, respectively.

For 1-d CA, amount of i cell ($1 \leq i \leq n$) at t is shown by $a_i(t)$ and is calculated by this formula:

$$a_i(t+1) = \varphi[a_{i-1}(t), a_i(t), a_{i+1}(t)]$$

In this formula, if φ is affected by the neighbors, it is general. If φ is a function of neighbor's cell collection and central cell, it is totalistic.

$$a_i(t+1) = \varphi[a_{i-1}(t) + a_i(t) + a_{i+1}(t)]$$

3. The Proposed Automatic Summarization Model

We have two modes of operations:

1. Training mode where features are extracted from 16 manually summarized English documents and used to train Cellular Automata, Fuzzy and Genetic programming models.

2. Testing mode, in which features are calculated for sentences from one English document. (These documents are different from those that were used for training.) The sentences are ranked according to the sets of feature weights calculated during the training stage. Summaries consist of the highest-ranking sentences.

3.1. Cellular automata model

The basic purpose of Cellular Automata (CA) is optimization. Since optimization problems arise frequently, this makes CA quite useful for a great variety of tasks. As in all optimization problems, we are faced with the problem of maximizing/minimizing an objective function f(x) over a given space X of arbitrary dimension [17]. Therefore, CA can be used to specify the weight of each text feature.

For a sentence s, a weighted score function, is exploited to integrate all the eight feature scores mentioned in Section 2, where w_i indicates the weight of f_i.

The Cellular Automata (CA) is exploited to obtain an appropriate set of feature weights using the 17 manually summarized English documents. A chromosome is represented as the combination of all feature weights in the form of w_i.

Thousand states for each iteration were produced. Evaluate fitness of each state (we define fitness as the average precision obtained with the state when the summarization process is applied on the training corpus), and retain the fittest 8 state to mate for new ones in the next iteration. In this experiment, thousand iterations are evaluated to obtain steady combinations of feature weights. A suitable combination of feature weights is found by applying CA. All document sentences are ranked in a descending order according to their scores. A set of highest score sentences are chronologically specified as a document summary based on the compression rate.

4. Experimental result

4.1. The English data

Seventeen English articles in the domain of science were collected from the Reading Book. Seventeen English articles were manually summarized using compression rate of 30%. These manually summarized articles were used to train the previously mentioned three models. The other one English article was used for testing. The average number of sentences per English articles is 85.8, respectively.

4.2. Cellular automata configuration

We are going to exploit the CA approach of [3], for summarization and use it as a baseline approach. For a sentence s, a weighted score function, is exploited to integrate the eight feature scores mentioned in previous

Related parameters for the training and testing of the CA model like States, Rules, Neighbor and other are given in Table 1 and 2.

Table 1: CA Data	
Independent Variables:	8
Training Samples:	1016
Testing Samples:	105

Table 2: CA General Settings	
2,3	States
256, 7625597484987	Rules
Von Neumann	Neighbor

4.3. The result of cellular automata model

We have exploited the CA approach of [11], for summarization as described above. Therefore, we have exploited the eight features for summarization. The system calculates the feature weights using Cellular Automata.

All document sentences are ranked in a descending order according to their scores. A set of highest score sentences are chronologically specified as a document summary based on the compression rate. To do CA concepts we using CA Classification model [11].

section, where w_i indicates the weight of f_i. We use the approach for testing; a set of 17 English documents was used. We apply f_i after using the defined weights from CA execution. All document sentences are ranked in a descending order according to their scores. A set of highest score sentences are chronologically specified as a document summary based on the compression rate.

4.3.1. CA model explicit formulation

By using CA rules and analyzing data we got set of rules are given in figure 1:

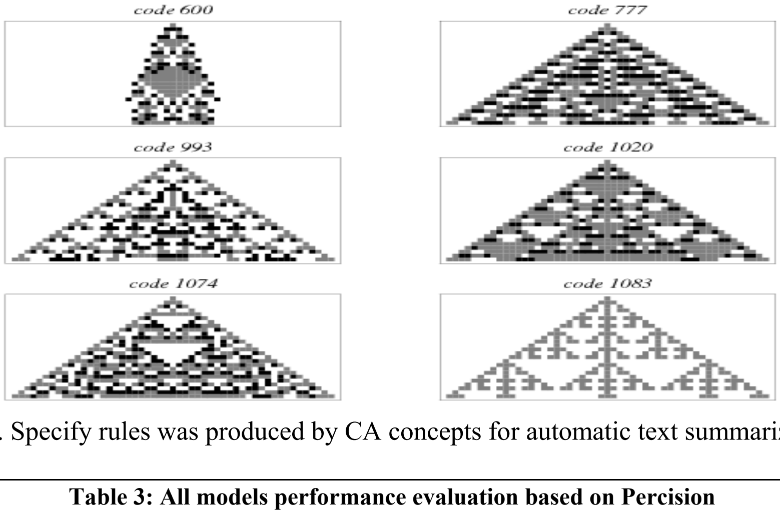

Fig.1. Specify rules was produced by CA concepts for automatic text summarization

Table 3: All models performance evaluation based on Percision			
Compression Rate (CR)	10%	20%	30%
	Precision (P)	Precision (P)	Precision (P)
CA Model	28.18%	29.04%	31.88%
Fuzzy Model	36.36%	42.86%	46.88%
Genetic Programming Model	54.54%	57.14%	59.38%

4.3.2 Evaluation CA model

We used 16 English text documents for training and one for testing CA model and the results are given in table 4 and 5:

Table 4: Statistics – Training	
Best Fitness:	679.43
Max. Fitness:	1000
Accuracy:	68.04%

Table 5: Statistics – Testing	
Best Fitness:	425.96
Max. Fitness:	1000
Accuracy:	43.81%

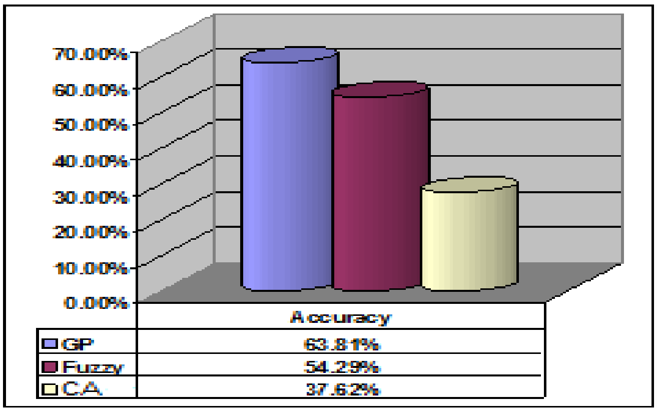

Fig.2: The accuracy for all models

4.3.3. Discussion

It is clear from Table 3 that this approach cannot be extended to the genre of newswire text. Fig.2 shows the total system performance in terms of precision for in case of all models for English articles, respectively. It is clear from the figure that CA approach gives the lowest results since CA has a bad capability to model arbitrary densities. The Fuzzy model and GP has better precision than the CA model.

5. Conclusion and Function Work

In this paper, we have investigated the use of Cellular Automata (CA), genetic programming approach and fuzzy approach for automatic text summarization task. We have applied our new approaches on a sample of 17 English scientific articles. Our approach results outperform the baseline approach results. Our approaches have been used the feature extraction criteria which gives researchers opportunity to use many varieties of these features based on the text type.

In the future work, we will extend this approach to multi-document summarization by addressing some anti-redundancy methods which are needed, since the degree of redundancy is significantly higher in a group of topically related articles than in an individual article as each article tends to describe the main point as well as necessary shared background.

6. References

[1] Aone, C., Gorlinsky, J., Larsen, B., and Okurowski, M. E. , *A Trainable Summarizer with Knowledge Acquried from Robust NLP Techniques*, Advances in Automatic Text Summarization, The MIT Press, Cambridge, Massachusetts, 1999, pages 71-80.

[2] Dehkordi, P., K., Khosravi, H. and Kumarci, F., *Text Summarization Based on Genetic Programming*, International Journal of Computing and ICT Research (IJCIR), Vol.3 No.1, June, 2009, pp. 57-64.

[3] Ferrier, L., *A Maximum Entropy Approach to Text Summarization*, School of Artificial Intelligence , Division of Informatics , University of Edinburgh, 2001.

[4] Harabagiu, S., Hickl, A., Lacatusu, F., *Satisfying information needs with multi-document summaries*, Information Processing & Management 43 (6), 2007, 1619–1642.

[5] Hirao, T., Okumura, M., Yasuda, N., Isozaki, H., *Supervised automatic evaluation for summarization with voted regression model*, Information Processing & Management 43 (6), 2007, 1521–1535.

[6] Hobson, S., Dorr, B., Monz, C., Schwartz, R., *Task-based evaluation of text summarization using relevance*

prediction, Information Processing & Management 43 (6), 2007, 1482–1499.

[7] Kyoomarsi.F , Khosravi.h., Eslami.E and Davoudi.M, *EXTRACTION-BASED TEXT SUMMARIZATION USING FUZZY ANALYSIS*. Iranian Journal of Fuzzy Systems Vol. 7, No. 3, (2010) pp. 15-32

[8] Lin, C., *Training a Selection Function for Extraction*, In the 8th International Conference on Information and Knowledge Management (CIKM 99), Kansa City, Missouri, 1999, 112-129.

[9] Luhn, H.P., *The automatic creation of literature abstracts*, IBM Journal of Research and Development 2 (2), 1958, 159–165.

[10] Moens, M., *Summarizing court decisions*, Information Processing & Management 43 (6), 2007, 1748–1764.

[11] Moore, A. (2003). *New Constructions in Cellular automata*, Oxford University Press.

[12] Neto, J. L., Freitas, A. A., and Kaestner, C. A. A., *Automatic Text Summarization using a Machine Learning Approach*, In Proc. 16th Brazilian Symp. on Artificial Intelligence (SBIA-2002). Lecture Notes in Artificial Intelligence 2507, Springer-Verlag, 2002. pp205-215.

[13] Neumann, V. (1996). *The Theory of Self-Reproducing Automata*, A. W. Burks (ed), Univ. of Illinois Press, Urbana and London.

[14] Nomoto, T., *Discriminative sentence compression with conditional random fields*, Information Processing & Management 43 (6), 2007, 1571–1587.

[15] Sekine, S. and Nobata, C. *Sentence Extraction with Information Extraction technique*, In Proc. Of ACM SIGIR'01 Workshop on Text Summarization. New Orleans, 2001, 1115-1129.

[16] Wolfram, E. (2002). A New Kind of Science, Wolfram Media, Inc.

[17] Yeh, S.J., Ke, T.H., Yang, M.W., Meng, L.I., ml_chg_old>Ye Yeh et al., *Text summarization using a trainable summarizer and latent semantic analysis*, Information Processing & Management 41 (1), 2005, 75–95.

[18] Zajic, D., Dorr, B., Lin, J., Schwartz, R., *Multi-candidate reduction: sentence compression as a tool for document summarization tasks*, Information Processing & Management 43 (6), 2007, 1549–1570.