AN IMPROVING THE EFFECTIVENESS OF TEXT TO MATRIX GENERATOR BY USING GENETIC ALGORITHM

Manish Sharma¹, Mr. Rahul Patel²

¹PG Scholar, CSE, AITR, Indore
²Assistant professor, CSE, AITR, Indore

ABSTRACT

Retrieval of relevant documents from a huge collection of documents is a traditional task. As genetic algorithms (GA) are efficient search and optimization techniques, they can be used to search the huge document search space. In this field, the aim of the GA is to help an IR system to find, in a huge documents text collection, a good reply to a query expressed by the user. In this paper, we investigate how to improve an efficiency of Text to Matrix Generator (TMG). Here, we are proposing a genetic approach in Text to Matrix Generator (TMG) to increase the efficiency. We present experimental results using the improvement in average number of terms.

Keywords: Information Retrieval (IR), Genetic algorithm (GA), Text to Matrix Generator (TMG), Vector Space Model (VSM).

1. INTRODUCTION

Information retrieval is generally considered as a subfield of computer science that deals with the representation, storage, and access of information [1]. Information retrieval is concerned with the organization and retrieval of information from large database collections [2]. Information Retrieval (IR) is the process by which a collection of data is represented, stored, and searched for the purpose of knowledge discovery as a response to a user request (query) [3]. This process involves various stages initiate with representing data and ending with returning relevant information to the user. Intermediate stage includes filtering, searching, matching and ranking operations. The main goal of information retrieval system (IRS) is to “finding relevant information or a document that satisfies user information needs”. To achieve this goal, IRSs usually implement following processes: In indexing process the documents are represented in summarized content form. In filtering process all the stop words and common words are remove. Searching is the core process of IRS. There are
various techniques for retrieving documents that match with users need. There are two basic measures for assessing the quality of information retrieval [2].

Precision: This is the percentage of retrieved documents that are in fact relevant to the query. Recall: This is the percentage of documents that are relevant to the query and were, in fact, retrieved. In this paper, we present an approach to improve the performance of text to matrix generator (TMG). Text to Matrix Generator (TMG) is a MATLAB Toolbox that can be used for various Data Mining (DM) and Information Retrieval (IR) tasks.

The structure of this paper is as follows. A brief literature review is presented in Section II, followed by vector space model in section III. Followed by genetic algorithm in Section IV. Followed by proposed method in section V, Followed by Experiment in section VI Followed by Result in section VII Finally, Section VIII covers conclusions.

2. RELATED WORK

Bangorn Klabbankoh and Ouen Pinngern [4] analyzed vector space model to boost information retrieval efficiency. In vector space model, IR is based on the similarity measurement between query and documents.


Mohammad Othman Nassar et al. [6] investigate Genetic algorithms to optimize the user query in the vector space model.


3. VECTOR SPACE MODEL

The vector space model can best be characterized by its attempt to rank documents by the similarity between the query and each document [10]. In the Vector Space Model (VSM), documents and query are represent as a Vector and the angle between the two vectors are computed using the similarity cosine function. Similarity Cosine function can be defined as:
where,

Documents and queries are represented as vectors.

$dl_j = \{w_{1,j}, w_{2,j}, \ldots, w_{t,j}\}$

$q = \{w_{1,q}, w_{2,q}, \ldots, w_{t,q}\}$

Vector Space Model have been introduce term weight scheme known as if-idf weighting. These weights have a term frequency (tf) factor measuring the frequency of occurrence of the terms in the document or query texts and an inverse document frequency (idf) factor measuring the inverse of the number of documents that contain a query or document term [4].

4. GENETIC ALGORITHM

Genetic Algorithm (GA) is a global optimization algorithm derived from evolution and natural selection. Although genetic algorithm cannot always provide optimal solution, it has its own advantages and is a powerful tool for solving complex problems.

Genetic algorithm is a powerful search mechanism and it is suitable for the information retrieval for the following reasons [18].

The document search space represents a high dimensional space. GAs are one of the powerful searching mechanisms known for its robustness and quick search capabilities. So they are suitable for information retrieval. In comparison with the classical information retrieval models, GA manipulates a population of queries rather than a single query. Each query may retrieve a subset of relevant documents that can be merged. The traditional methods of query expansion manipulate each term independent of other. GA contributes to maintain useful information links representing a set of terms indexing the relevant documents. The traditional methods of relevance feedback are not efficient when no relevant documents are retrieved with the initial query.

Genetic algorithm operations can be used to generate new and better generations. The genetic algorithm operations include:

4.1 Reproduction: the selection of the fittest individuals based on the fitness function.

4.2 Crossover: is the exchange of genes between two individual chromosomes that are reproducing. In one point cross over a chunk of connected. Genes will be swapped between two chromosomes.

4.3 Mutation: is the process of randomly altering the genes in a particular chromosome. There are two types of mutation:

4.3.1 Point mutation: in which a single gene is changed.

4.3.2 Chromosomal mutation: where some number of genes is changed completely.
As shown in figure 1 a simple GA works as follows:

1. Start with a randomly generated population.
2. Evaluate the fitness of each individual in the population
3. Select individuals to reproduce based on their fitness
4. Apply crossover
5. Apply mutation
6. Replace the population by the new generation of individuals
7. Go to step 2.

5. PROPOSED METHOD

5.1 Research Objective

The aim of this proposed work is to retrieve the relevant documents by using the best combination of the term list, given a set of document collections. The terms that are extracted from the document collections for generating the combination terms after obtaining the best combination of terms, it is applied to the information retrieval system to obtain more relevant documents. Genetic Algorithm enhances average numbers of terms in Text to Matrix Generator by optimize the objective function.

5.2 Proposed approach

The keywords extracted from the document collections are stored in the database. A frequency measure is associated with each keyword. Here we are using Genetic Algorithm approach to obtain the best combination of keywords by optimizing the fitness function to enhance the
effectiveness of Text to Matrix Generator. The advantage of the proposed approach improves the effectiveness of TMG and retrieves the more relevant document when a query is given. In this paper, we are proposing Genetic Algorithm in Text to Matrix Generator toolbox for improving the performance of the system.

5.3 Methodology

We are performing the GA operation in Text to Matrix Generator on MATLAB simulator. For the optimization of objective function we use optimization toolbox of MATLAB simulator to get the results.

6. EXPERIMENT

In this section we discuss about how experiment is conducted and result occur during experiment. In our experiment we put objective function of TMG in optimization tool. A series of computer experiments was conducted in order to evaluate proposed GA enabled Text to Matrix Generator (TMG). Experiments were executed using data taken from the TMG collection. In this the collection was indexed, which is based on normalized term frequency and normalized inverse document frequency. Indexed collection contained 12 documents and 549 numbers of terms.

7. RESULTS

Adding new Keyword and Calculating Improvement:

<table>
<thead>
<tr>
<th>S.No</th>
<th>Avg. no. of terms before GA</th>
<th>Avg. no. of terms after GA</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>164.417</td>
<td>173.0010986328125</td>
<td>8.58409</td>
</tr>
</tbody>
</table>

8. CONCLUSION

In this paper, we have investigated how to improve efficiency of Text to Matrix (TMG). We have proposed a Genetic Algorithm in Text to Matrix (TMG) to improve retrieval effectiveness. Genetic algorithm is an excellent optimization tool. The algorithm use fitness function which is represented by the equation gives more sophisticated result.

By using genetic algorithm in Text to Matrix Generator (TMG) the number of Iterations will be increases which will improve the performance.

REFERENCES


