Oriental Journal of Computer Science & Technology Vol. **3**(2), 227-231 (2010)

Mining intelligent E-voting data: A framework

JULIUS O. OKESOLA[1], OLUWAFEMI S. OGUNSEYE[2], KAZEEM I. RUFAI[1] and OLUSEGUN FOLORUNSO[2]

[1]Department of Computer & Information Sciences,
Tai Solarin University of Education, Ijebu-Ode, Ogun State (Nigeria).
[2]Department of Computer Science, University of Agriculture, Abeokuta, Ogun State (Nigeria).

(Received: July 26, 2010; Accepted: September 01, 2010)

ABSTRACT

Intelligent e-voting data has been shown to pose a lot of benefit to e-voting especially in the area of security and recounting. After the election and balloting processes, valuable knowledge can still be extracted from this data. This work provides a framework model as roadmap for developers to follow in future development of such a system. The Perl based sample tested showed optimum performance and hence proves the viability of the methodology.

Key words: Text Mining, e-voting, knowledge extraction, data mining, semantic data, tree traversal.

INTRODUCTION

Extracting knowledge through the mining of raw data has been of major interest for researchers, system developers, business managers and the corporate world as a whole (Sumathi & Sivanandam, 2006). The reason for this is not farfetched, data mining helps in the production of knowledge and the discovery of new patterns to describe the data. This implies hidden or subtle patterns can be discovered. This is useful in forecasting and predicting future behavior of systems.

Most applications of data mining has however focused on the mining of raw and ordinary data as against intelligent data proposed by Delphi(2004). Research on the use of intelligent/ annotated/semantic data have shown that it can be used in web mining and can also be mined (Stumme *et al.,* 2005). The mining of semantic data and its use in mining the web is an area that is under very active research. Issues that have been encountered include standardization of the parts of the semantic web e.g. the data representation format (Stumme *et al.,* 2005).

Semantic data involves the annotation of a key data with a body of other data describing it or giving further information about it. Semantic annotation formally identifies concepts and relations between concepts in documents and is intended primarily for use by machines (Uren *et al.,* 2005). Semantic annotation enhances information retrieval and improve interoperability. Information retrieval is improved because of the ability to link to other valuable source documents. Its structure is synonymous to a tree and can therefore be processed as such. The key data has branches that leads to other data (nodes). This is illustrated in figure 1 below.

Uren *et al.,* (2005) conducted a survey where they sufficiently described methods of annotating data. Ogunseye *et al.,* (2010) used one of these methods in annotating ballot data in their paper on an e-voting visual analytics system. In this work, we show a framework for extracting useful knowledge from the e-voting data used in Ogunseye *et al.,* (2010). The primary purpose of which is to generate knowledge that can be useful after an election process such as for planning and decision making by all the stake holders of electioneering.

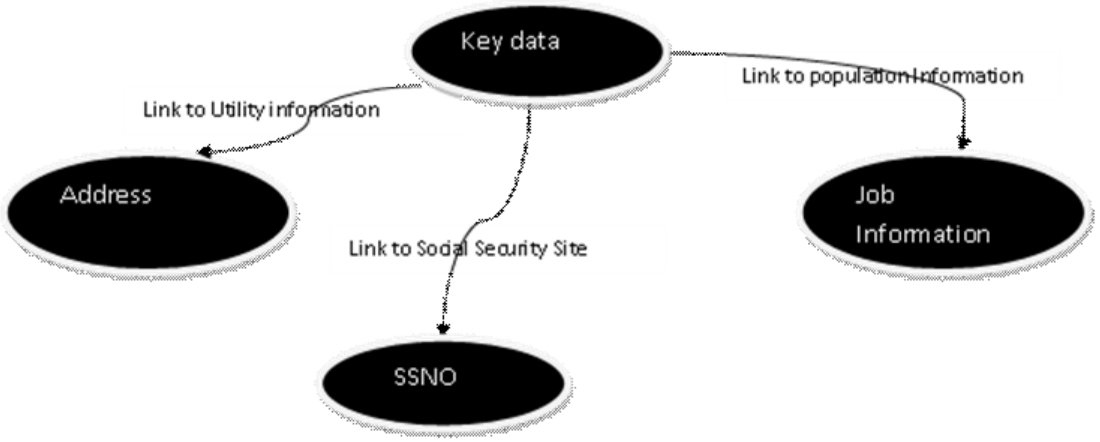

Fig. 1: annotation links from an intelligent data tree

Because of the nature of the data in question, traditional query and data mining would not be efficient or appropriate (Tulder,2003), we therefore apply a leverage – tree traversal to an aspect of data mining – text mining to get the Job done. This work therefore provides a framework that developers can build on for mining intelligent data and indeed extracting knowledge from e-voting system data residue/byproduct.

The diagram shown in figure 1 shows how the key data which implies the main data used in the election e.g. voter's name is linked to other data or data sources.
Sample e-voting data used in Ogunseye et al (2010) is shown below

<voter-e-voting-system-ontology: voting data
rdf: ID ="#Zone3 ward 19"/>
<active-e-voting-system-ontology: has-name
rdf:resource "#oluwafemi shawn ogunseye"/>
<active-e-voting-system-ontology: has-SSN
rdf:resource "#NGABK001"/>
<active-e-voting-system-ontology: has-sex
rdf:resource "#M"/>
<voter-e-voting-system-ontology : verification data
rdf: wsid:"http://www.nigeriassn.gov/private/12399/
"/>
<active-e-voting-system-ontology: has-name
rdf:resource "#folorunso olusegun"/>
<active-e-voting-system-ontology: has-SSN
rdf:resource "#NGABK023"/>
<active-e-voting-system-ontology: has-sex
rdf:resource "#M"/>
<voter-e-voting-system-ontology : verification data
rdf: wsid:"http://www.nigeriassn.gov/private/66399/
"/>
<active-e-voting-system-ontology: has-name
rdf:resource "Julius Okeosola"/>
<active-e-voting-system-ontology: has-SSN
rdf:resource "#NGLAG001"/>
<active-e-voting-system-ontology: has-sex
rdf:resource "#M"/>
<voter-e-voting-system-ontology : verification data
rdf: wsid:"http://www.nigeriassn.gov/private/12452/
"/>

The data above stores vital information about the voters and also stores the verification site address.
The annotations provide the information used in the search.

Method

In order to successfully mine the data, it is treated as a tree with nodes and a root. The step taken to mine it is divided into two, these are:

1. Traversing the tree {Web mining}
2. Text mining the nodes {Semantic Mining}

Traversing the Tree (Intelligent data)

The method presented here is based on previous usage mining techniques and algorithms (Stumme *et al.,* 2005) , where semantic annotation has been used in finding out how users browse the

web studying how they move from link to link & enabling them come back to previous links on subsequent visits to the site. However we do not apply it to usage mining but to locate information in linked documents that will be text mined.

Let kd, (x_1,y_1) (x_2,y_2)….. (x_n,y_n) where kd = key data and (x_i,y_i) indicates that the voters information x_i is found in y_i and $y_i = x_{i+1}$. Converting this to a traversal tree we add a new vertex y_i into the tree together with (x_i,y_i). The key data is the starting point and from there we traverse the tree. The tree could therefore look like this. <(null, A),(A,B), (B,C),> here the tree has two nodes apart from the root. B & C. There is no need to traverse any node twice in this usage. This is illustrated in Fig. 2 below.

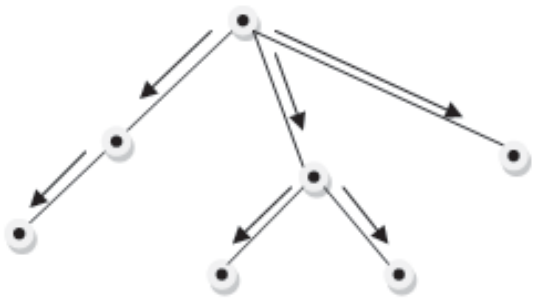

Fig. 2: The traversal flow of the intelligent data

Text Mining the node

The method proposed here involve the application of Text mining processes on collection of words and phrases that are extracted from documents. The documents are processed as follows:

Tokenization – this involves using white spaces and punctuation to identify lexical items in the text. A simple rule based approach can be applied to extract the terms of interest. The approach proposed here ia a thinned down version of what Brill (1995) used in forming morpho-syntactic categories and tagging parts of speech in his work.

Filtering – A filtering module is included to ensure accuracy. The filtering module is simply rule based with rules pseudo coded as follows:

```
IF Name != text then Flag “error”;
IF Age != integer then Flag “error”;
IF Location != exist() then flag “error”

Traverse(node) // starts at the root node
 while hasleftchild(node) do
  node = node.left
 do
  mine(node) // each node is mined herein
  if (hasrightchild(node)) then
   node = node.right
   while hasleftchild(node) do
    node = node.left
  else
      while node.parent ≠ null and node =
node.parent.right
     node = node.parent
    node = node.parent
 while node ≠ null
```

The algorithm for the framework model based on the explanation already given above is as follows:

Sample Implementation

In the prototype implementation which we used in the evaluation of our model and design, the steps discussed above were observed.

The model was tested on Linux platform because of its amenability to scripting with Perl and the use of the console. Perl is a scripting language that can provide access to Linux commands, allow string or text manipulation and afford other benefits.

The traversal was done in Perl and each link was downloaded using The w3m console based application and the content of the linked page is directed to a file called “rawkontents” using the “>” syntax of linux. The rawkontent file is immediately piped to our perl mining script file to be mined and the extracted text is directed to a file called the “minedkontent”.

For the sake of this sample only frequency was checked from our data.

The result of the a trial mine for sex distribution in a fictitious data plotted into chart for easy cognition is shown in Fig. 4.

```
File  Edit  Format  View  Help
print OUTPUT "Sum of Male\n";
close(OUTPUT);
die ("Can't open $outputfile for reading\n")
unless open(INPUT, "< $outputfile");
while (<INPUT>) {
chomp;
if(defined $_) {
print STDOUT "$_\n";
}
}
close(INPUT);
#unlink($outputfile);
```

Fig. 3: Sample Perl mining script file

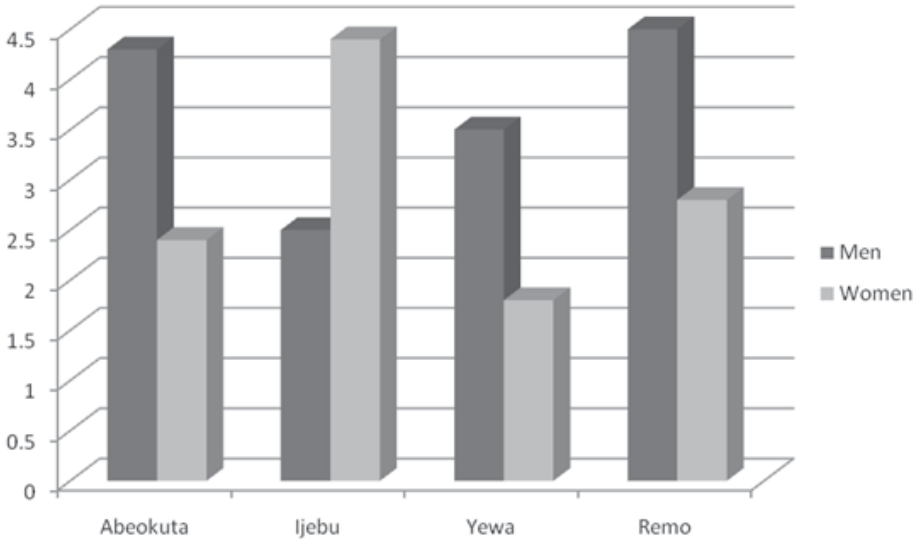

Fig. 4: Showing the results of mining a fictitious senatorial ballot data. Voters (x100) against Senatorial district

DISCUSSION

From our experiment in Fig. 4 it was observed that the system has an accuracy level of 98.2% and that it omitted data that mainly had typographical errors like 'i0' instead of '10' which resulted in the lack of perfection in accuracy. The implementation is however basic and can become more complex with increase in data size and data complexity. The overall performance of the application was viewed using the 'top' command in Linux and it was not too CPU intensive with a CPU usage of 22%.

The methodology presented in this work is generalized to allow for easy modification and improvement where possible. The work provides a framework through which knowledge from e-voting data can be extracted. This work will serve as a guide for the future development of mining for e-voting data and can also show how to mine intelligent data in general. This knowledge can have far-reaching effect on electioneering stakeholders. These stakeholders which include but is not limited to the government, the electoral commission and political parties will use the knowledge derived from such a system to plan and strategize for subsequent election. Government and electoral bodies can use it to budget resources that will be needed in the next election. Stakeholders can decipher which age group to reach out to in order to encourage participation amongst other strategic planning that can be done. It will also emphasize transparency in the balloting process.

REFERENCES

1. Delphi Group, The document is the process, White Paper, Delphi Consulting Group Inc., http://www.delphigroup.com/research/whitepapers/DocIsProcess.pdf (1994).
2. Uren Victoria, Cimiano Philipp , Iria Jos´e , Handschuh Siegfried, Vargas-Vera a Maria, Motta a Enrico, Ciravegna Fabio (2005), “Semantic annotation for knowledge management: requirements and a survey of the state of the art”Journal of web semantics, Elsevier.
3. Brill E., "Transformation-based error-driven learning and natural language processing: A case study on the part-of speech tagging" computational linguistics, **21**(4): 543-565 (1995).
4. Ogunseye O.S. Folorunso O. Okesola J.O. Woodward J.R., “The EVAS MODEL: Solving E-voting Problems in Nigeria”, Oriental Journal of Computer and Information Technology, India (2010) [In press].
5. S. Sumathi, S.N. Sivanandam, “Introduction to Data Mining and its Applications”, Studies in Computational Intelligence, 29 (2006)
6. Stumme G., Hotho A., AND BERENDT B., “Semantic Web Mining State of the Art and Future Directions”, available online at: http://test.websemanticsjournal.org/openacademia/p (2005)