Chapter 6

6. The Implementation

6.1 Introduction

This chapter describes the implementation details of this system. The implementation can be categorized into four main sections as the implementation of the SQL Server analysis services application for data mining, web reporting application, database implementation tools and the data mining implementations. Following section demonstrates the implementation details of each section of this system with their processes

6.1 The Questionnaire

Questionnaire is already sent with the acknowledgement of Area Engineer in Jaffna to the area office in Jaffna and the meter readers are distributed to the consumers that when they go for the meter reading. For the next bill cycle the questionnaire is collected.

Meanwhile they conducted the telephone interviews when the consumers make inquiries to the help desk. And also the questionnaire is distributed to the consumers who are visited to the CEB Premises. Comparing the first method of distribution questionnaire through meter readers this telephone interviewing was not succeeded as expected.

6.2 Consumption and Billing Related Data Extraction

The Billing and Consumption data in domestic sector of Jaffna area for one year is extracted from the billing database and the consumers are categorized in to different average consumption units. For example

- Consumers with average consumption less than 60 units
- Consumers with average consumption more than 60 and less than 90
- Consumers with average consumption more than 90 and less than 120
- Consumers with average consumption more than 120 and less than 180
- Consumers with average consumption greater than 180

6.3 Analysis Services Application for data mining

The SQL Server Analysis Services application is used for the implementation of data mining application. For that SQL Server is having in build Integrated Development Environment called SQL Server Business Intelligence Development Studio is used. There you can create projects for connecting SQL Server database to do data mining. Below figure shows my sample application interface at the implementation stage of the system.

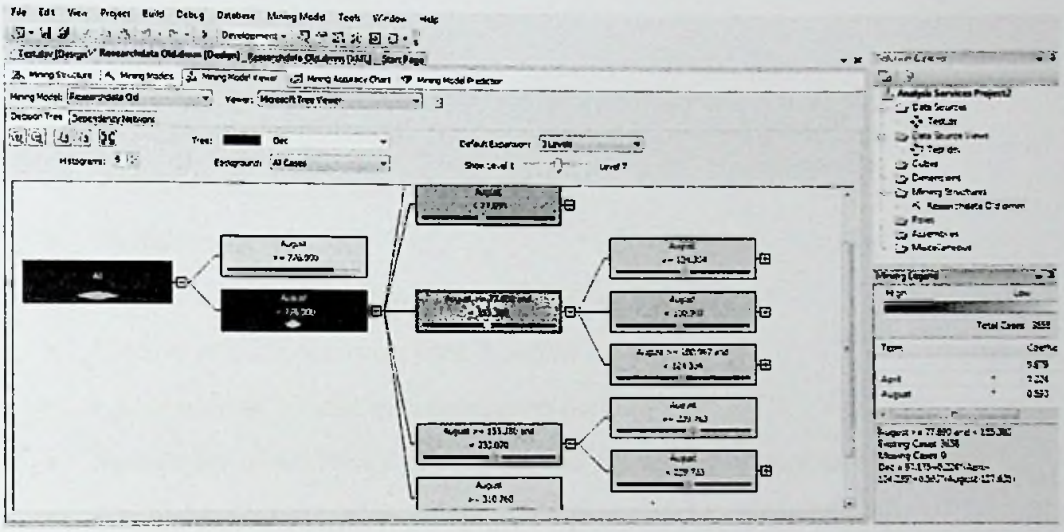

Figure 12: SQL Server Business Intelligent Interface

Here first you need to create a data source from the database columns that you need to extract the patterns and the relationship among those selected attributes. Then you need to add this data source to the data source view where the place you are applying data mining algorithms. Then you can apply data mining algorithms for that data source view. In SQL Server Analysis Services there are different kinds of data mining algorithms existing. The algorithms are

1. Microsoft Association Rules
2. Microsoft Clustering
3. Microsoft Decision Trees
4. Microsoft Linear Regression
5. Microsoft Logistic Regression
6. Microsoft Naïve Bayes
7. Microsoft Neural Networks

8. Microsoft Sequence Clustering
9. Microsoft Time Series

Among those I have basically used Microsoft Decision Tree, Microsoft Clustering and Microsoft Naïve Bayes classification algorithms for my analysis as my research is basically consisting of classifications of consumers in different ways of using electricity.

Selecting Attributes for the data mining

- The occupation
- Monthly Income
- No of people in the house
- The floor area of the house
- Education Level
- Electric Appliances usage pattern (Behavior Pattern)
- Electric Appliances usage time duration
- Knowledge of electricity conservation methods
- Knowledge of electricity calculating and the tariff structure (awareness)
- The highest consumption month of the year and the reasons for that

The implemented application can be hosted in a server and allow people to browse the application in such a manner the consumption pattern can be extracted.

6.4 Web Reporting Application

Web reporting application is implemented using Visual Studio 2012 Development Enviromnet.Here I have used C# (ASP.NET) programming language for the development of the application. That application is having a user login where users are redirected to the specific area related profiles. This method of login will be more important at the expansion of the application to other areas in the country. There is a facility for a user to enter data from the questionnaire directly to the database. A web based data entry form is integrated with this application where administrators of the application can login and enter data for specific area. Hence the future improvements can be done in areas separately.

The application is basically implementing reports, charts, prediction reports, mertixes where you cannot extracted only from applying mining models. From simple to complex analysis are also included this application. This application can be used for average awareness person of computer literacy in Ceylon Electricity Board for take analysis reports when necessary. Some source codes and Graphical user interfaces are shown below. Rest of the source codes and Graphical User Interfaces are in Appendix of the thesis.

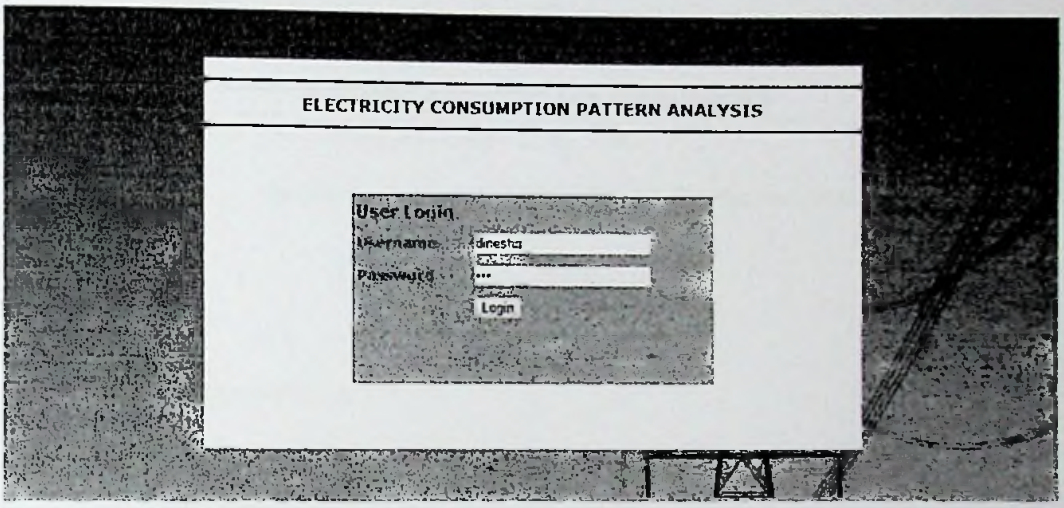

Figure 13: Login page of the web application

Source code for Login

```
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.Sql;
using System.Data.SqlClient;
using System.Data.OleDb;
using System.Data;

public partial class Login : System.Web.UI.Page
{
    SqlDataAdapter adapter;
    SqlCommandBuilder cmdBuilder;
    DataSet ds = new DataSet();
    DataSet changes;

    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        Logins(TextBox1.Text, TextBox2.Text, lblarea.Text);
    }
```

```
    public bool Logins(string username, string password,string area_code)
    {
        bool isOk = false;
        try
        {
            string sql = "";

           SqlConnection connection = ConStrings.GetConnection();
                connection.Open();
                sql = "select password from users where username='" + username
+ "' and active_st='Y' and area_code='" + area_code + "'";

                adapter = new SqlDataAdapter(sql, connection);
                adapter.Fill(ds);
                connection.Close();

                if (ds.Tables[0].Rows.Count > 0)
                {
                    Response.Redirect("individualConsumption.aspx");
                }



            }

        catch (Exception ex)
        {
        }

        return isOk;
    }
}
```

Source Code for Individual Electricity Usage Pattern analysis

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using Microsoft.Reporting.WebForms;

public partial class IndividualUsagePattern : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void btnView_Click(object sender, EventArgs e)
    {
        AverageConsumption(account_number.Text.Trim());
        AverageCharge(account_number.Text.Trim());
    }

    private DataTable CreateDataTable()
    {
```

```
        DataTable myDataTable = new DataTable();
        DataColumn myDataColumn;

        myDataColumn = new DataColumn();
        myDataColumn.DataType = Type.GetType("System.String");
        myDataColumn.ColumnName = "Month";
        myDataTable.Columns.Add(myDataColumn);

        myDataColumn = new DataColumn();
        myDataColumn.DataType = Type.GetType("System.Decimal");
        myDataColumn.ColumnName = "Consumption";
        myDataTable.Columns.Add(myDataColumn);

        return myDataTable;
    }

    private void AddDataToTable(string unit, decimal count, DataTable myTable)
    {
        DataRow row;
        row = myTable.NewRow();
        row["Month"] = unit;
        row["Consumption"] = count;
        myTable.Rows.Add(row);
    }


    public DataSet AverageConsumption(string acc)
    {
        string month="";
        string sql = "";
        DataSet ds = new DataSet();
        DataTable mytable=CreateDataTable();

        try
        {

            using (SqlConnection connection = ConStrings.GetConnection())
            {

                sql = "select
January,February,March,April,May,June,July,August,September,October,November,De
cember from individualConsumptions where account_number='"+acc+"' ";

                SqlCommand cmd = new SqlCommand(sql, connection);
                cmd.CommandType = CommandType.Text;
                SqlDataAdapter ad2 = new SqlDataAdapter(cmd);
                ad2.Fill(ds, "consumption");

                for (int i = 0; i < 12; i++)
                {
                    if(i==0)
                    {
                        month="January";
                    }
                    if(i==1)
                    {
                        month="February";
                    }
                    if(i==2)
                    {
                        month="March";
                    }
```

```
                    if(i==3)
                    {
                        month="April";
                    }
                    if(i==4)
                    {
                        month="May";
                    }
                    if(i==5)
                    {
                        month="June";
                    }
                    if(i==6)
                    {
                        month="July";
                    }

                    if(i==7)
                    {
                        month="August";
                    }
                    if(i==8)
                    {
                        month="September";
                    }
                    if(i==9)
                    {
                        month="October";
                    }
                    if(i==10)
                    {
                        month="November";
                    }
                    if(i==11)
                    {
                        month="December";
                    }

                    int count = int.Parse(ds.Tables[0].Rows[0][i].ToString());
                    AddDataToTable(month, count,mytable);
                }

                ReportDataSource datasource = new ReportDataSource("DataSet1",
mytable);
                ReportViewer1.LocalReport.DataSources.Clear();
                ReportViewer1.LocalReport.ReportPath =
"ConsumptionByAccountNumber.rdlc";
                ReportViewer1.LocalReport.DataSources.Add(datasource);



            }
        }

        catch (Exception ex)
        {
        }

        return ds;
    }

    public DataSet AverageCharge(string acc)
```

```
    {
        string month = "";
        string sql = "";
        DataSet ds = new DataSet();
        DataTable mytable = CreateDataTable();

        try
        {

            using (SqlConnection connection = ConStrings.GetConnection())
            {

                sql = "select c1,c2,c3,c4,c5,c6,c7,c8,c9,c10,c11,c12 from
individualConsumptions where account_number='" + acc + "' ";

                SqlCommand cmd = new SqlCommand(sql, connection);
                cmd.CommandType = CommandType.Text;
                SqlDataAdapter ad2 = new SqlDataAdapter(cmd);
                ad2.Fill(ds, "consumption");

                for (int i = 0; i < 12; i++)
                {
                    if (i == 0)
                    {
                        month = "January";
                    }
                    if (i == 1)
                    {
                        month = "February";
                    }
                    if (i == 2)
                    {
                        month = "March";
                    }
                    if (i == 3)
                    {
                        month = "April";
                    }
                    if (i == 4)
                    {
                        month = "May";
                    }
                    if (i == 5)
                    {
                        month = "June";
                    }
                    if (i == 6)
                    {
                        month = "July";
                    }

                    if (i == 7)
                    {
                        month = "August";
                    }
                    if (i == 8)
                    {
                        month = "September";
                    }
                    if (i == 9)
                    {
                        month = "October";
```

```
                    }
                    if (i == 10)
                    {
                        month = "November";
                    }
                    if (i == 11)
                    {
                        month = "December";
                    }


                    decimal charge =
decimal.Parse(ds.Tables[0].Rows[0][i].ToString());
                    AddDataToTable(month, charge, mytable);


                }

                ReportDataSource datasource2 = new ReportDataSource("DataSet2",
mytable);
                ReportViewer1.LocalReport.ReportPath =
"ConsumptionByAccountNumber.rdlc";
                ReportViewer1.LocalReport.DataSources.Add(datasource2);



            }
        }

        catch (Exception ex)
        {
        }

        return ds;
    }
}
```

Source Code analyze Electricity Equipments using inside a house

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using Microsoft.Reporting.WebForms;

public partial class equipmentswithconsumption : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void btnView_Click(object sender, EventArgs e)
    {
        DataTable mydatatable = CreateDataTable();
        int start = 0;
        int end = 0;
        string unit = "";
        if (ddlunits.SelectedValue.Equals("2"))
```

```
        {

             unit = "0-30";
             start = 0;
             end = 30;
        }

     if   (ddlunits.SelectedValue.Equals("3"))
        {
            start =31;
            end = 60;
             unit = "31-60";
        }
         if (ddlunits.SelectedValue.Equals("4"))
        {
            start =61;
            end = 90;
             unit = "61-90";
        }

         if (ddlunits.SelectedValue.Equals("5"))
         {
             start = 91;
             end = 120;
             unit = "91 to 120";
         }
         if (ddlunits.SelectedValue.Equals("6"))
         {
             start = 121;
             end = 180;
             unit = "121 to 180";
         }
         if (ddlunits.SelectedValue.Equals("7"))
         {
             start =181;
             end = 350000;
             unit = "above 180";
         }

            DataSet ds = Equipments(start,end, "refregerator");
            if (ds.Tables[0].Rows.Count > 0)
            {

                for (int i = 0; i < ds.Tables[0].Rows.Count; i++)
                {
                    AddDataToTable(unit,
int.Parse(ds.Tables[0].Rows[i][0].ToString()), "Refrigerator",
ds.Tables[0].Rows[i][1].ToString(), mydatatable);
                }
            }


            DataSet ds1 = Equipments(start,end, "Iron");
            if (ds1.Tables[0].Rows.Count > 0)
            {

                for (int i = 0; i < ds1.Tables[0].Rows.Count; i++)
                {
                    AddDataToTable(unit,
int.Parse(ds1.Tables[0].Rows[i][0].ToString()), "Iron",
ds1.Tables[0].Rows[i][1].ToString(), mydatatable);
                }
```

```
            }
            DataSet ds2 = Equipments(start, end, "heater");

            if (ds2.Tables[0].Rows.Count > 0)
            {
                for (int i = 0; i < ds2.Tables[0].Rows.Count; i++)
                {
                    AddDataToTable(unit,
int.Parse(ds2.Tables[0].Rows[i][0].ToString()), "Heater",
ds2.Tables[0].Rows[i][1].ToString(), mydatatable);
                }
            }
            DataSet ds3 = Equipments(start, end, "kettle");

            if (ds3.Tables[0].Rows.Count > 0)
            {
                for (int i = 0; i < ds.Tables[0].Rows.Count; i++)
                {
                    AddDataToTable(unit,
int.Parse(ds3.Tables[0].Rows[i][0].ToString()), "Kettle",
ds3.Tables[0].Rows[i][1].ToString(), mydatatable);
                }
            }
            DataSet ds4 = Equipments(start, end, "waterpump");

            if (ds4.Tables[0].Rows.Count > 0)
            {
                for (int i = 0; i < ds.Tables[0].Rows.Count; i++)
                {
                    AddDataToTable(unit,
int.Parse(ds4.Tables[0].Rows[i][0].ToString()), "Water Pump",
ds4.Tables[0].Rows[i][1].ToString(), mydatatable);
                }
            }
            DataSet ds5 = Equipments(start,end, "electricoven");

            if (ds5.Tables[0].Rows.Count > 0)
            {
                for (int i = 0; i < ds5.Tables[0].Rows.Count; i++)
                {
                    AddDataToTable(unit,
int.Parse(ds5.Tables[0].Rows[i][0].ToString()), "Electric Oven",
ds5.Tables[0].Rows[i][1].ToString(), mydatatable);
                }
            }
            DataSet ds6 = Equipments(start, end, "ricecooker");

            if (ds6.Tables[0].Rows.Count > 0)
            {
                for (int i = 0; i < ds6.Tables[0].Rows.Count; i++)
                {
                    AddDataToTable(unit,
int.Parse(ds6.Tables[0].Rows[i][0].ToString()), "Rice Cooker",
ds6.Tables[0].Rows[i][1].ToString(), mydatatable);
                }
            }
            DataSet ds7 = Equipments(start, end, "blender");

            if (ds7.Tables[0].Rows.Count > 0)
            {
                for (int i = 0; i < ds7.Tables[0].Rows.Count; i++)
                {
```

```
                    AddDataToTable(unit,
int.Parse(ds7.Tables[0].Rows[i][0].ToString()), "Blender",
ds7.Tables[0].Rows[i][1].ToString(), mydatatable);
                }
            }
            DataSet ds8 = Equipments(start, end, "fans");

            if (ds8.Tables[0].Rows.Count > 0)
            {
                for (int i = 0; i < ds8.Tables[0].Rows.Count; i++)
                {
                    AddDataToTable(unit,
int.Parse(ds8.Tables[0].Rows[i][0].ToString()), "Fans",
ds8.Tables[0].Rows[i][1].ToString(), mydatatable);
                }
            }
            DataSet ds9 = Equipments(start, end, "tv");
            if (ds9.Tables[0].Rows.Count > 0)
            {
                for (int i = 0; i < ds9.Tables[0].Rows.Count; i++)
                {
                    AddDataToTable(unit,
int.Parse(ds9.Tables[0].Rows[i][0].ToString()), "Television",
ds9.Tables[0].Rows[i][1].ToString(), mydatatable);
                }
            }
            DataSet ds10 = Equipments(start, end, "computer");
            if (ds10.Tables[0].Rows.Count > 0)
            {
                for (int i = 0; i < ds10.Tables[0].Rows.Count; i++)
                {
                    AddDataToTable(unit,
int.Parse(ds10.Tables[0].Rows[i][0].ToString()), "computer",
ds10.Tables[0].Rows[i][1].ToString(), mydatatable);
                }
            }




            ReportDataSource datasource = new ReportDataSource("DataSet1",
mydatatable);

            ReportViewer1.LocalReport.DataSources.Clear();
            ReportViewer1.LocalReport.ReportPath = "NoOfQuipmentsByRange.rdlc";
            ReportViewer1.LocalReport.DataSources.Add(datasource);
        }



    private DataTable CreateDataTable()
    {
        DataTable myDataTable = new DataTable();
        DataColumn myDataColumn;

        myDataColumn = new DataColumn();
        myDataColumn.DataType = Type.GetType("System.String");
        myDataColumn.ColumnName = "unit";
        myDataTable.Columns.Add(myDataColumn);
```

```
        myDataColumn = new DataColumn();
        myDataColumn.DataType = Type.GetType("System.Int32");
        myDataColumn.ColumnName = "count";
        myDataTable.Columns.Add(myDataColumn);

        myDataColumn = new DataColumn();
        myDataColumn.DataType = Type.GetType("System.String");
        myDataColumn.ColumnName = "equipment";
        myDataTable.Columns.Add(myDataColumn);

        myDataColumn = new DataColumn();
        myDataColumn.DataType = Type.GetType("System.String");
        myDataColumn.ColumnName = "status";
        myDataTable.Columns.Add(myDataColumn);



        return myDataTable;
    }
    private void AddDataToTable(string unit, int count, string equipment,string
status, DataTable myTable)
    {
        DataRow row;
        row = myTable.NewRow();
        row["unit"] = unit;
        row["count"] = count;
        row["equipment"] = equipment;
        row["status"] = status;



        myTable.Rows.Add(row);
    }


    public DataSet Equipments(int start,int end,string equipment)
    {

        string sql = "";
        DataSet ds = new DataSet();
        DataTable mydatatable = CreateDataTable();
        try
        {

            using (SqlConnection connection = ConStrings.GetConnection())
            {

                    sql = "select count("+equipment+"),"+equipment+" from
researchdata where avg_consumption > = " + start + " and avg_consumption
<="+end+"  group by "+equipment+"";
                    SqlCommand cmd = new SqlCommand(sql, connection);

                cmd.CommandType = CommandType.Text;
                SqlDataAdapter ad2 = new SqlDataAdapter(cmd);
                ad2.Fill(ds, "consumption");

            }

        }
```

```
        catch (Exception ex)
        {
        }

        return ds;

    }

}
```

The rest of the source codes, Graphical User Interfaces and the outputs are attached in the Appendix section.

6.5 Microsoft Excel Add in for SQL Server 2008 for data mining

This is also a very powerful tool for data mining. At the starting of my application implementation I have used this add in for Excel 2007.Here all you need to do is create a connection to the SQL Server 2008 instance. The connection makes the bridge between Excel and the database. So you can use each and every data mining algorithm in Excel itself to analyze your data. At the implantation stage I have used this tool as well. Below figure shows a sample of a cluster diagram derived from that tool.

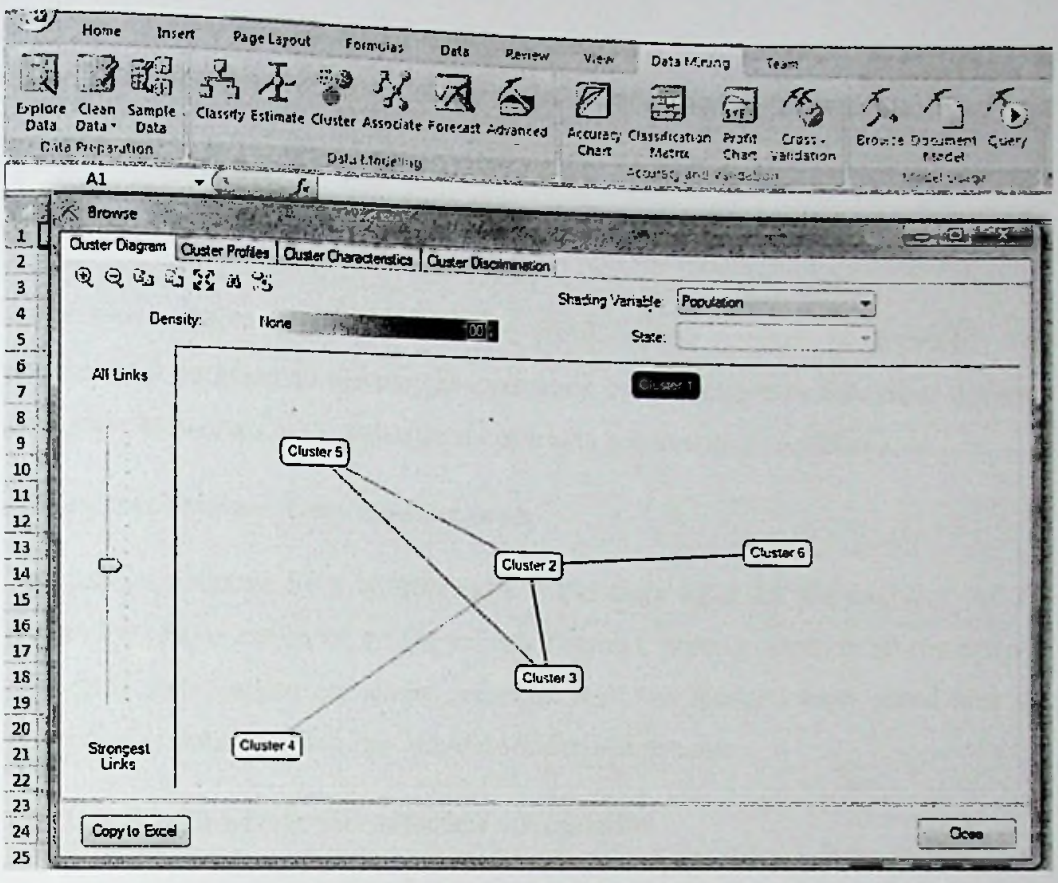

Figure 14: Microsoft Excel Add in for SQL Server 2008 for data mining

6.6 Summary

This chapter summarizes the implementation details of different components and interrelation among them. How the each module of the system is implemented is also described in detail. The implementation is very close relationship with the design of the system because I have tried so many possibilities for design the system and within that I have chosen this design is the most appropriate way of implementing the application in a sophisticated manner. The next chapter will describe the evaluation of the system against objectives. The main Objective of my research is to identify electricity consumption patterns of households and identify the crucial factors affecting for electricity consumption as well as electricity conservation.