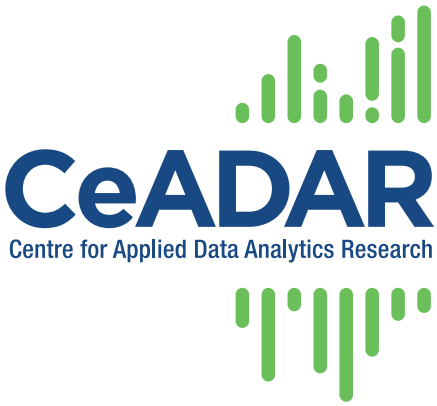

CeADAR – Centre for Applied Data Analytics Research

Enterprise Ireland Data Analytics Technology Centre

SmartAd – Weak Signals and Causal Relationships – Technical Specification

Document Type:	Technical Specification
Project Title:	CeADAR
DDN Theme:	1 – Intelligent Analytic Interfaces
Theme Leader:	Sarah Jane Delany (DIT)
DDN Sub-Theme:	Ease of Interaction
Authors:	Ingo R. Keck, Susan McKeever, Sarah Jane Delany
Document Version:	0.1
Date of Delivery to ISG:	February, 2015
Number of pages:	6

ABSTRACT

In this report we provide a brief Technical Specification of the prototype for the SmartAd – Weak Signals and Causal Relationships project, part of the CeADAR Ease of Interaction theme

CeADAR is a research partnerhip comprising University College Dublin, University College Cork, and Dublin Institute of Technology.

http://www.ceadar.ie

Contents

1 Description of Industry Needs

One of the major questions that arose from our discussions with companies involved in the advertising space is how they can determine which advertising initiatives have been successful and which have not. This age-old question in advertising gave rise to the famous John Wanamaker quote: "Half the money I spend on advertising is wasted; the trouble is, I don't know which half".

The state of the art in marketing research so far is to analyse the direct impact of different advertisement campaigns separately. In reality, consumers are not just subject to the influence of one campaign, instead they are constantly presented with different forms of advertisements via a subset or all of the noted channels. The effects of campaigns in different channels might interact: A customer that has first seen an advertisement on the TV and heard about it on the radio might be more easily influenced to register a corresponding ad on a social network that finally leads him to the offer webpage of the company.

Further more, the problem is not unique to the advertising space. Companies regularly want to see how events in different channels or signals interact or influence each other. It is possible to move away from the advertising name space and towards a more general technology that focusses on data fusion and estimation of mutual correlation. This way this project can be used to analyse generic signals, as long as they are similar to the ones observed in advertisement campaigns, regarding their statistical characteristics.

2 System(s) Involved

The main systems involved in this project is the data provided by the company partners and the actual front- and backend of the demonstrator.

2.1 Data

For this project we plan to use example data sets provided by Omnicom Media for advertisement campaigns and by Danutech for DSL line quality.

Source	Omnicom Media	Danutech
Type	Marketing Campaigns	DSL quality for single lines
Channels	few (up to 10)	many (up to 4000)
Base channel	click-throughs or sales	overall signal quality
Cause channels	ad events	SNR/QLN/Bitwith per frequency bin
Temporal resolution	seconds to days	minutes
Pre-Processing	none/PCA	PCA (high mutual correlation)

Table 1: Data to be analysed

The data will be saved locally and can then be used in the demonstrator. An additional testing dataset will be created for public demonstrations, as the other two data sets are only for internal evaluation.

2.2 Front- And Backend

The analysis will be performed by a Python process. A WSGI web frontend will be provided to select the data, set up the analysis, and allow a basic visualisation of the result in a web browser.

3 Approach

There are some fundamental assumptions in the use of the demonstrator that have to be taken into account:

- time series data of all channels is available
- overlapping time frames of all channels
- the format of the input data matches the requirement set out in the Appendix of this document.

The workflow of SmartAd is shown in Figure 1. It consists of multiple steps that are described in the following sections.

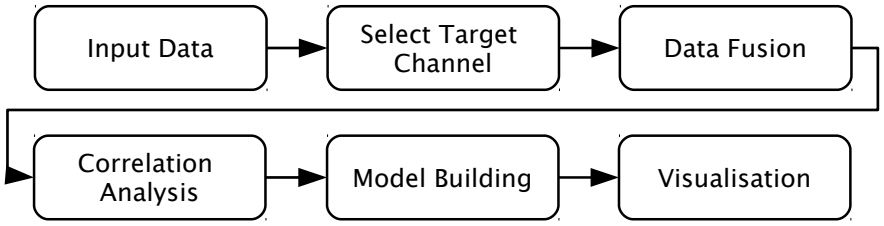

Figure 1: The SmartAd Workflow consists of several steps, as pictured in this figure.

3.1 Data Input And Target Channel Selection

In the first step, the data is loaded into the demonstrator. To be able to work with the data, basic information has to be contained in the data set, like the time points of data points and the type of the channel (to pre-select typical models in the fusion step). The appendix of this report contains a detailed definition of the proposed data format.

This project tries to find correlations between 'influencer' channels (e.g. advertisement data) and a 'target' channel (e.g. click-through rates, sale rates, etc). It is therefore necessary to select the one channel that contains the 'target'.

3.2 Data Fusion

For the data fusion a common time frame for the analysis has to be selected by the user. Channels with a higher temporal resolution will then be interpolated downward, while for channels with a lower temporal resolution, a fusion model has to be defined that allows the creation of an intermediate signal for this channel with the necessary resolution. The selection of the correct parameters for this fusion model is crucial for the accuracy of the data analysis. Expert user input will be necessary to do this. The demonstrator will include parameters for simple fusion modelling to illustrate how channels with different temporal resolutions can be synchronised.

Data sets may contain hundreds of highly correlated channels that increase the memory and time requirements of the analysis without providing additional information. By using Principal Component Analysis (PCA) this number can be reduced to a manageable set of channels, reducing the noise in the data set and restricting the analysis to the underlying dimension of the data set. PCA may also combine mutually correlated channels to channel clusters. This is expected to happen in the DSL data set.

3.3 Data Analysis

We will use Granger causality for the correlation analysis to determine whether a time lag in the data gives higher correlation between the target channel and the influencer channels. If the Granger causality results do not indicate a significant time lag, the time lags will not be used in the subsequent model building stage.

In a next step a model will be built to identify the influence of the influencer channels on the target channel. The results of the data analysis will give the impact of certain influencer channels on the target channel relative to other influencer channels.

The parameters of the previous data fusion step will directly influence the final result of this data analysis, therefore actual numeric outcomes from the analysis will be reported as relative impact values, rather than actual weights of interactions.

3.4 Presentation Of Result

The visualisation will focus on showing a connection graph which will illustrate the strength of the connections between the channels and the probability or certainty of the estimation of the connection strength. It will also show the meta data of the channels available if the user is interested in it.

Figure 2 shows such a graph in the basic view. The strength of connections is represented by line widths. The probabilities or certainty of the connection strengths are shown by the intensity of the line. Connections that show a low probability of estimation in the analysis will not be plotted at all.

Figure 3 shows a generic example visualisation of the result we expect for the DSL data set. Various influencer signals have been clustered together by the PCA step. In this example the user has clicked on the top right cluster and gets additional information about this influencer.

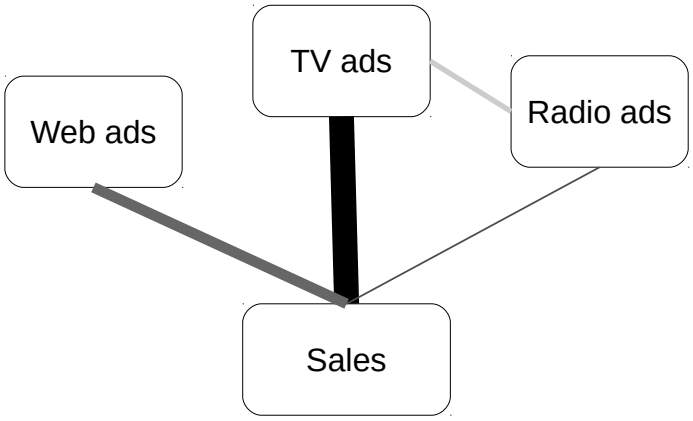

Figure 2: Example visualisation of the connection between different channels in an ad campaign data set. The width of the line corresponds to the strength of the connection, the intensity represents the probability of the estimation of the connection strength.

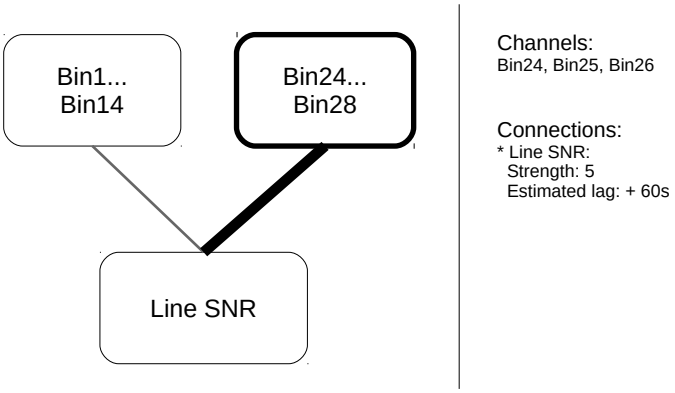

Figure 3: Example visualisation of the result for the DSL data set. The user has clicked on the channel cluster "Bin24...Bin28", so the interface gives additional information on this cluster.

4 Appendix

We define the input data format for the demonstrator in the following sections.

4.1 Input Data Format

The information given in table 2 is necessary for the analysis and has to be contained in the data set.

The data should be provided in the JSON format (UTF-8 encoded as default). The main container should be a list, where each list item is a dictionary using the requirements from table 2 as keys. Data values and data time points should be lists of the respective values. Date and time should be given as "yyyy-mm-ddThh:mm:ss±hh:mm".

An example file is given in figure 4.

Requirement	*Format*	*Content*
channel title	text	descriptive short title of the channel
channel info	text	description of the channel
target channel	boolean	1 for target channels, 0 for influencer channels
channel type	text	type of the channel, i.e. tv ads, web ads, etc
data unit	text	unit of the provided data (for example "views", "impressions", "sales")
data additive	boolean	1 for data that adds up between data points (e.g. sales, views), 0 for data that does not add up (e.g. temperature, SNR, etc)
data values	numeric	list of numeric values of the data points
data time points	date and time as text	list of corresponding times of the respective data values from the data values list

Table 2: Input data requirements

```
 [{"channel title":  "TV",
"channel info":"TV advertisements in the 2013 campaign",
"target channel" :  0,
"channel type":  "tv ads",
"data unit":  "ad views",
"data additive" :  1,
"data values" :  [10000, 10000, 10000],
"data time points":  ["2011-12-04T00:00:00+00:00",
"2011-12-08T00:00:00+00:00", "2011-12-10T00:00:00+00:00"] },
{"channel title":  "Sales",
"channel info":"Sales information in the 2013 campaign",
"target channel" :  1,
"channel type":  "sales",
"data unit":  "sales",
"data additive" :  1,
"data values" :  [20, 20, 34, 50],
"data time points":  ["2011-12-03T18:00:00+00:00",
"2011-12-05T18:00:00+00:00", "2011-12-08T18:00:00+00:00",
"2011-12-10T18:00:00+00:00"] }]
```

Figure 4: Example JSON data file. The file should contain all data channels as a JSON list, where each list item is a separate channel, given as a JSON dictionary. Data values and data time points are also JSON lists of values and time strings