Visual Bayesian fusion to navigate a data lake

Conference Paper · July 2016

7 authors, including:

Karamjit Singh
Tata Consultancy Services Limited
15 PUBLICATIONS 14 CITATIONS
SEE PROFILE

Kaushal Paneri
Northeastern University
5 PUBLICATIONS 8 CITATIONS
SEE PROFILE

Garima Gupta
TCS Research India
11 PUBLICATIONS 11 CITATIONS
SEE PROFILE

Geetika Sharma
Tata Consultancy Services Limited
23 PUBLICATIONS 53 CITATIONS
SEE PROFILE

Visual Bayesian Fusion to Navigate a Data Lake

Karamjit Singh, Kaushal Paneri, Aditeya Pandey, Garima Gupta, Geetika Sharma, Puneet Agarwal, Gautam Shroff
TCS Research
Email: {karamjit.singh, kaushal.paneri, aditeya.pandey, gupta.garima1, geetika.s, puneet.a, gautam.shroff}@tcs.com
Tata Consultancy Service Ltd., Gurgaon India

***Abstract*—The evolution from traditional business intelligence to big data analytics has witnessed the emergence of 'Data Lakes' in which data is ingested in raw form rather than into traditional data warehouses. With the increasing availability of many more pieces of information about each entity of interest, e.g., a customer, often from diverse sources (social-media, mobility, internet-of-things), fusing, visualizing and deriving insights from such data pose a number of challenges: First, disparate datasets often lack a natural join key. Next, datasets may describe measures at different levels of granularity, e.g., individual vs. aggregate data, and finally, different datasets may be derived from physically distinct populations. Moreover, once data has been fused, queries are often an inefficient and inaccurate mechanism to derive insight from high-dimensional data. In this paper we describe iFuse, a data-fusion based visual analytics platform for navigating a data lake to derive insights. We rely on Bayesian graphical models to provide useful rudder with which to fuse and analyze disparate islands of data in a systematic manner. Our platform allows for rich interactive visualizations, querying and keyword-based search within and across datasets or models, as well as intuitive visual interfaces for value-imputation or model-based predictions. We illustrate the use of our platform in multiple scenarios, including two public data challenges as well as a real-life industry use-case involving the probabilistic fusion of datasets that lack a natural join-key.**

I. Introduction

Traditional business intelligence is rapidly evolving to adopt modern big-data analytics architectures based on the concept of a 'data lake', where, rather than first integrating multiple historical data from diverse sources into a common star schema via extraction-transformation-load operations, the datasets are maintained in their raw form for analysis and visualization, as also described in our prior work [1]. The data-lake approach becomes essential when datasets are not only sourced from intra enterprise transaction systems, but also from social-media, Internet-of-things, surveys, etc., often making their integration into one common schema a challenging or even impossible task. Further, insights generated from previously sourced data can also be loaded as a new dataset in the data-lake. Such an environment leads to a number of challenges while deriving analytical insights; for example, dealing with incongruous join keys between different datasets [1].

In this paper, we illustrate the process of navigating and deriving insights from diverse datasets in a data-lake architecture while enumerating the challenges to be addressed at each stage, along with solutions based on Bayesian model inference and advanced data visualisation. We narrate the steps, challenges and their solution using our data-lake platform in the context of two data-challenges, which our group has won or was selected in top-5 submissions, as well as a real-world data integration scenario faced in practice.

In practice, when faced an analytical question to be answered using data, the first and foremost challenge is to search and browse the datasets available in one's data-lake. For locating the dataset of interest, we also need to see summary of the datasets available. We employ advanced data visualization methods to see the individual datasets, so that users could check the quality of data, degree of sparseness etc. before choosing a dataset. Thereafter we need to identify potential join keys between target datasets for our target analytical task. Sometimes these join keys can be simple and straightforward to perform the join, at other times the potential join keys may not have the same level of granularity. For example, in a data-lake created using the data from data.gov.in (hereafter referred to as disease control data-lake, see Section III), one of the dataset reports state-wise death-rates for various diseases, while another dataset contains level of pollutants at different river-beds in those states, or it might contain only the name of the river and its river-bed. (In the latter scenario, i.e., where the state-names were not available we could probabilistically join the datasets based on geographical maps, using Bayesian fusion of data as presented in our prior work [1]; Note: such map-based fusion is not used in this paper though.)

Alternatively, sometimes a join-key does not exist between two datasets due to master-data mismatch between organizations resulting in, say, different item codes used in different geographies for the same product. For example, in the real-world data-lake created for one of our enterprise customer (hereafter referred to as Product data-lake, see Section V), one of the datasets contains the global-product-ids used at the global-level along with their key-characteristics. The local-product-ids along with their key characteristics used in various countries were present in another dataset. Here, the local-product-ids for the same product were different for every country as well as different from global-product-ids, as a result no mapping existed between same product from two countries. As a result, it becomes hard to compare the sales of a portfolio of products sold in two different countries. In such cases our platform uses a probabilistic join based on product characteristics, employing an ensemble of Bayesian network models and textual similarity as described in Section V.

Another frequently observed issue, especially when dealing with raw data from diverse sources, is that of missing values. For example, in our product data-lake, not all product characteristics are present for every product, and we may need to fill these for the purpose of visual analytics. Missing value imputation can also be viewed as that of predicting a value based on a probabilistic model, such as is often desired, i.e., predict labels or values in a target dataset based on models learned from training data: Our platform uses the same Bayesian network models for imputation, value prediction and probabilistic join.

Last but not least, a common requirement while answering analytics questions is that of "what-if analysis", where an analysts queries one or more target measures under specified conditions. We illustrate the use of Bayesian inference to answer both "what-if queries" as well as predicting salaries using two novel visualizations of Bayesian inference, both of which we use in the context of second data challenge on analyzing labor market presented in Section IV.

The key contributions of this paper are: i) enumerating the challenges involved in deriving analytical insights from data in a data-lake architecture ii) real-world results on probabilistic fusion of datasets that do not share a natural join-key, using iii) a novel ensemble-based probabilistic join based on confidence scores for picking the model (Bayesian network or similarity search); iv) a novel visual depiction of Bayesian imputation for filling missing data and value prediction using parallel co-ordinates; v) our novel visualization of conditional queries powered by Bayesian inference; and finally vi) an end-to-end procedure to navigate and derive insights from data in a data-lake, illustrated in the context of real-world enterprise data-lake as well as two data-challenges, and implemented in a common platform that integrates the techniques i-v in a novel manner.

The remainder of this paper is organized as follows: We begin with an overview of our platform, called iFuse, in Section II, which integrates the novel elements enumerated above. We next go on to illustrate our procedure for navigating a data-lake as well as describe each of these features while narrating the use of our iFuse platform in the context of the COMAD[1] data-challenge, which we we won, in Section III; in the context of CoDS[2] data challenge, where we were in top-5 entries, in Section IV; and finally in the context of a real-world enterprise data-lake in Section V. Finally, we conclude in Section VII, after a brief description of related work in Section VI.

II. Navigating a Data Lake: iFuse Overview

As already mentioned above: traditional business intelligence is enabled by large data warehouses having carefully curated datasets using star-schemas, whereas data-lakes have a more flexible organization of constituent datasets leaving a choice with analysts for picking the suitable dataset for their analysis. This makes it possible for the analysts to attempt newer type of analysis than those the business intelligence infrastructure was originally designed for. The *iFuse* platform has been designed in a manner that it not only solves this aspect, but also includes suitable features such that the challenges highlighted in previous section get addressed. We now describe the features of *iFuse* platform.

Looking: An analyst navigating data residing in a data-lake (a) needs to be aware of the datasets available (b) should be able to search for datasets when required and (c) the platform should enable them to 'see' different datasets. In *iFuse* each dataset is represented visually as a 'data tile' consisting of a word cloud of tags generated from attribute names or provided by the user and the visualizations associated with the dataset. Users can flip a tile to see the complete list of data attributes and can select the tiles or its attributes for various purposes. We create an inverted-index [2] of the column-names of the datasets and additional keywords entered by a user at the time of uploading a dataset in the data-lake. As a result, users can search for a dataset by entering keywords of their choice. Finally, *iFuse* provides a number of multidimensional data visualizations such as motion charts and parallel coordinates [3] and spatial data visualizations such as cartograms and bubble maps to visually analyze datasets. All visualizations open in a 'Compare View' facilitating analysis using multiple charts.

Linking: Once appropriate datasets have been identified by the user, they may need to be joined to create a more meaningful dataset for analysis. *iFuse* supports joining datasets with a common key with multiple options such as inner, outer, left or right join. It provides a simple interface allowing users to select attributes from flipped tiles of multiple files and joining using the options available.

Finding Relationships: Dependencies between attributes from different datasets may be discovered using Bayesian Networks. In *iFuse* users can choose multiple attributes from different datasets, and add them to an attribute-cart. After they select one variable as the target variable, they can request for a network to be created, and the iFuse platform automatically creates a suitable network using minimum spanning tree (MST) based approach described later in Section IV. Once a network is created it gets recognized as a new dataset, which is then added into the inverted-index, and starts appearing as a tile on the home page. This network may then be used for further analysis.

Making Inferences: Bayesian networks may be used to make inferences and for 'what-if' analysis. *iFuse* provides what we call a 'Linked Query View' for the same. A 'Linked View' in data visualization parlance refers to a view containing multiple charts such that interaction with any chart synchronously updates all. We use a linked view to visualize conditional queries on attributes in the network.

Missing Data: Very often data values may be missing for certain attributes of constituent datasets in the data-lake, reducing the utility of such datasets. *iFuse* provides data completion feature using Bayesian imputation. Attributes that are part of a Bayesian network may be imputed in data sets where they are partially missing. For this users can select a network based tile to be used for imputation as well as the

[1] COMAD'16 - Conference on Management of Data, 2016

[2] CoDS'16 - ACM iKDD Conference on Data Science, 2016

dataset with missing values. A new dataset with complete data is generated, included in the inverted-index, and made available in *iFuse* for further visual analysis.

Prediction: Often attributes may need to be predicted for new datasets in which they do not exist at all. The Bayesian imputation feature of *iFuse* may be used to address this problem as well. The interface for prediction is the same as that for data completion described above.

Probabilistic Join: Data from disparate sources may not have a common join key however their fusion may be warranted for analytical reasoning. We introduce the concept of probabilistic joining of data. This pertains to the example described in Section I with respect to missing mapping between global-product-id from one dataset and local-product-id from another. We achieve this by predicting the global-product-id for every local-product-id using an ensemble of Bayesian prediction and textual similarity.

In subsequent sections, we illustrate the above features of *iFuse* in the context of three different data-lakes. Features related to **looking**, i.e., searching for data and exploratory visual analytics and **linking** are introduced in the context of the COMAD[1] data challenge on disease control. **Finding relationships**, **making inferences** and **missing data/prediction** are illustrated via the iKDD CODS[2] data challenge on analysing labor markets, and **probabilistic join** is essential for analysing disparate real-world product data.

III. Disease Control Data Lake

The first data lake consists of data from COMAD data-challenge[1] on disease control. Following is a brief description of the datasets made available as part of the data-challenge.

1) **Health data** Numbers of deaths and cases of various diseases such as Malaria, Diarrhoea and Japanese Encephalitis, in different states of India for multiple years.
2) **Expenditure data** Details of funds allocated by the government for the disease control through various schemes for multiple years.
3) **Water Pollution data** Water quality parameters such as temperature, pH and Nitrate, for major rivers and land habitations in India.

Some of questions posed by the data challenge required analyzing prevalence of specific diseases in various states of India, the relationship between water-borne diseases and water pollution and the effectiveness of efforts made to control diseases, e.g., through health-related government expenditure.

Add/Discover Datasets: A user may upload new dataset into *iFuse*. As part of the data-upload process, the new dataset gets included in the inverted-index with the tags automatically generated from column headers or additional keywords provided by the user. This enables keyword-based search on datasets for subsequent access and exploration. Datasets are visually represented as tiles on the search page, as shown in Figure 1, with word clouds of tags generated from attribute lists. The tiles can be 'flipped' by double-clicking, to see a complete list of attribute names and other meta data.

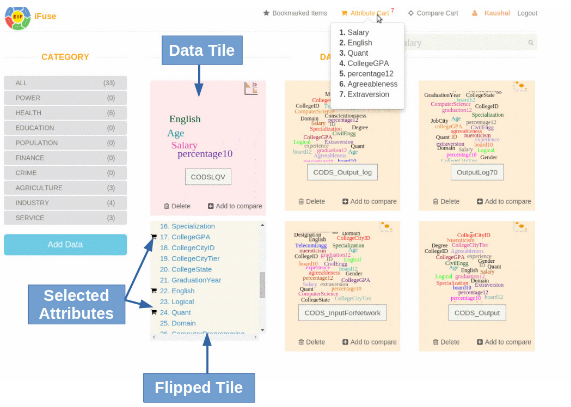


Fig. 1. Search page with Data Tiles, Flipped Tiles and Attribute Selection for Network Creation

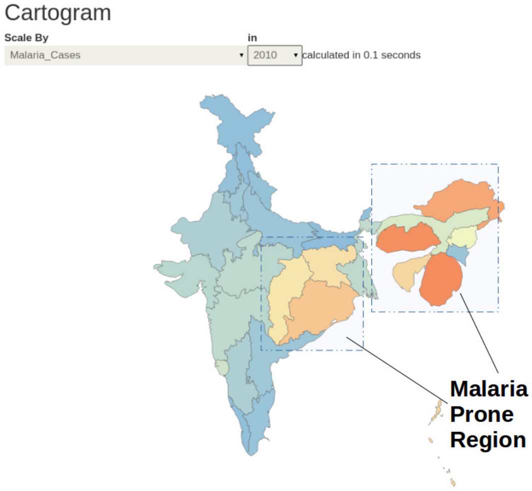
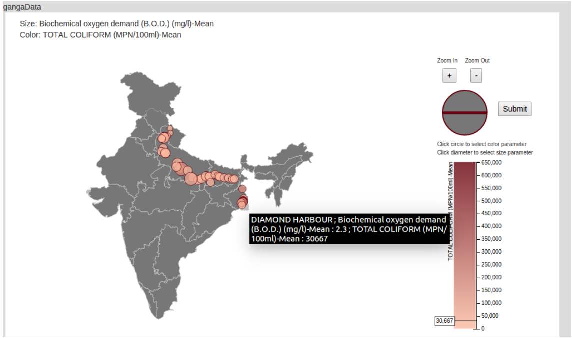


Fig. 2. Cartogram showing prevalence of Malaria

Exploratory Visual Analysis As most of the datasets in this data-lake are have spatial attributes, we used two map visualizations of *iFuse*- Cartograms and Bubble maps to show details of the datasets. Cartograms visualize quantitative data about regions such as countries and states, using colour variation and rubber-sheet distortions in area proportional to data values. As shown in Figure 2 we map data about Malaria on a cartogram to study its prevalence across India.

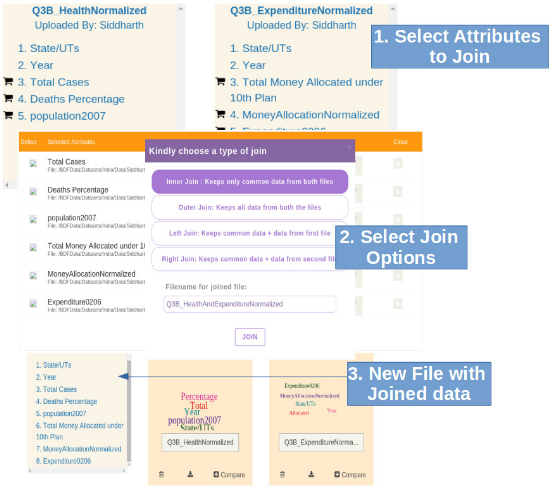
Bubble maps visualize data about specific geographical locations on a map with data attributes mapped to properties of bubbles or circles such as size and colour. For example, river pollution data was visualized using Bubble maps as shown in Figure 3. Here, color of the bubbles indicates 'Total Coliform (MPN/100ml)-Mean' and the diameter of the bubbles indicates 'Biochemical Oxygen Demand(mg/l)-Mean'. From this chart one can observe path followed by the river, and how pollution levels change as it flows from its source to the sea.



Size: Biochemical oxygen demand (B.O.D.) (mg/l)-Mean
Color: TOTAL COLIFORM (MPN/100ml)-Mean

DIAMOND HARBOUR ; Biochemical oxygen demand (B.O.D.) (mg/l)-Mean : 2.3 ; TOTAL COLIFORM (MPN/100ml)-Mean : 30667

Fig. 3. Bubble Map showing data for Ganga River


Q3B_HealthNormalized
Uploaded By: Siddharth

1. State/UTs
2. Year
3. Total Cases
4. Deaths Percentage
5. population2007

Q3B_ExpenditureNormalized
Uploaded By: Siddharth

1. State/UTs
2. Year
3. Total Money Allocated under 10th Plan
4. MoneyAllocationNormalized

1. Select Attributes to Join

Select	Selected Attributes	Close
	Total Cases File: /BDFData/Datasets/India/Data/Siddhart	
	Deaths Percentage File: /BDFData/Datasets/India/Data/Siddhart	
	population2007 File: /BDFData/Datasets/India/Data/Siddhart	
	Total Money Allocated under 1(File: /BDFData/Datasets/India/Data/Siddhart	
	MoneyAllocationNormalized File: /BDFData/Datasets/India/Data/Siddhart	
	Expenditure0206 File: /BDFData/Datasets/India/Data/Siddhart	

Kindly choose a type of join

Inner Join : Keeps only common data from both files
Outer Join: Keeps all data from both the files
Left Join: Keeps common data + data from first file
Right Join: Keeps common data + data from second fil

Filename for joined file:
Q3B_HealthAndExpenditureNormalized

JOIN

2. Select Join Options

1. State/UTs
2. Year
3. Total Cases
4. Deaths Percentage
5. population2007
6. Total Money Allocated under 10th Plan
7. MoneyAllocationNormalized
8. Expenditure0206

3. New File with Joined data

Fig. 4. Joining Two Datasets

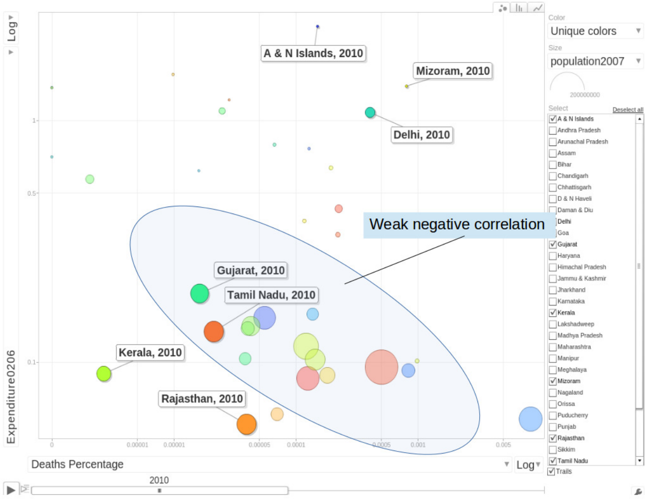


Fig. 5. Weak negative correlation between Expenditure and Deaths

We used the simple data join feature of *iFuse* to join datasets with Expenditure on health and total number of deaths, as shown in Figure 4. We plotted the joined data using a motion chart 5, and observed that there is a weak negative correlation between expenditure and percentage of deaths.

IV. Labour Markets Data Lake

The second data lake consists of data from CoDS data-challenge[2] on understanding the labour markets. This had data with profiles of engineers (degree, year of completion, domain, job designation, salary) as well as their scores on a standardised exam, which tested their logical, quantitative, English language and domain skills. Two important tasks posed by the data-challenge were to (a) predict salaries of candidates present in a test dataset and (b) recommend what changes in profiles would be required for obtaining higher salaries. We use the Bayesian networks capabilities of *iFuse* to accomplish these tasks.

Model Learning: Probabilistic dependence of data attributes may be learned using Bayesian networks. A Bayesian network is a graphical structure that allows representation and reasoning about an uncertain domain. It is a representation of joint probability distribution (JPD) which consist of two components. The first component G is directed acyclic graph whose vertices correspond to random variables. The second component, the conditional probability table (CPT), describes the conditional distribution for each variable given its parent-nodes. A CPT of a node indicates the probability that the each value of a node can take given all combinations of values of its parent-nodes. Considering a BN consisting of N random variables $X = (X_1, X_2, ..., X_N)$, the general form of joint probability distribution of a Bayesian network can be represented as in Equation 1, which encodes the BN property that each node is independent of other nodes, given its parents, where $Pa(X_i)$ is the set of parents of X_i.

$$P(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P(X_i \mid Pa(X_i)) \qquad (1)$$

We select top-K features from the dataset based on the mutual information of all features with target variable. Mutual information between continuous-continuous, and continuous-discrete variables is calculated using non-parametric entropy estimation toolbox (NPEET)[4]. This tool implements an approach presented by Steeg in [5] to find mutual information estimators, which is based on entropy estimates from k-nearest neighbor distances.

After feature selection, and learning the mutual information between all pairs of features, we obtain an undirected graph of attributes, referred to as feature graph. We learn efficiently executable Bayesian network from this graph, which includes a node for top-K features and the target variables. We call it Minimum Spanning Tree Network (MSTN). We learn the structure of MSTN with the following approach, given top-K features including both continuous and discrete variables :

1) We find a minimum spanning tree(MST) on feature graph, created by calculating pairwise mutual information between various features, and dropping the edges that have this mutual information smaller than a threshold.

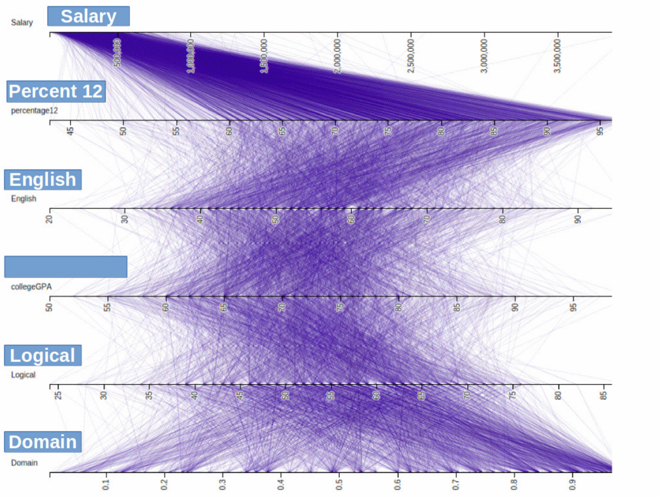


Fig. 6. Network showing dependence of Salary on other attributes.

2) We then initialize each edge to random direction.
3) We flip each edge direction to compute 2^{K-1} directed graphs and calculate the cross entropy of each graph.
4) We then choose a graph with least cross entropy.

After learning the structure of the Bayesian network in this manner, we learn the parameters of the network, i.e., CPTs for each node in a network.

iFuse users can learn a Bayesian network by selecting relevant attributes from different datasets joined inside the system. We provide a visual interface for this as shown in Figure 1. Data attributes for model learning can be selected from flipped data tiles and are added to the attribute cart. The user then chooses the 'Request Network' option, selects a target variable and triggers the network learning module in the back-end. It returns with the least cross entropy network which is visualized as a parallel coordinates plot with axes drawn horizontally in topological order.

In the Labour Market challenge, the task was to discover what affects salary from among a number of variables such as percentage of marks at high school and college level and scores on various tests. A network with salary, high school percentage, college GPA, English, Logical ability and Domain scores was discovered. A visualization of this network is shown in Figure 6.

Make Inferences: Once a network has been saved, it can be used to perform visual model inferencing using what we call a 'Linked Query View', shown in Figure 7. This is an interactive linked view especially designed to query Bayesian networks. The user selects n attributes from the network to query and these are visualized in an $n \times n$ chart grid with attributes repeated horizontally and vertically. Charts along the diagonal, show the probability distributions of the corresponding attribute as bar charts, in Figure 7. In the cells above the diagonal, we show scatter plots of the data with row and column attributes on the x and y axis of the plot, respectively. These provide a view of the data used to build the network and can be used to analyze pair-wise correlations between attributes.

In order to query the network, users can select ranges for multiple attributes by clicking on appropriate bars in the bar charts. This puts a condition on the attribute to be in the range selected by the user. On hitting the query button, a conditional query is executed on the network using Bayesian inference. The conditional distributions of the other attributes are computed and the bar charts are updated accordingly. We provide a comparison view with the initial and conditional distributions overlayed in the different colors so that changes in the distributions can be perceived easily.

We used the model inferencing feature of *iFuse* to answer the second question about recommendations to candidates, posed by the data-challenge. This required recommending the ideal candidate profile required to ensure a high salary. We designed a 'Bayesian Student Advisor' to provide recommendations using the Linked Query View shown in Figure 7. The original probability distributions are shown on the diagonal using bar charts with yellow bars. Salary is plotted on log_{10} scale.

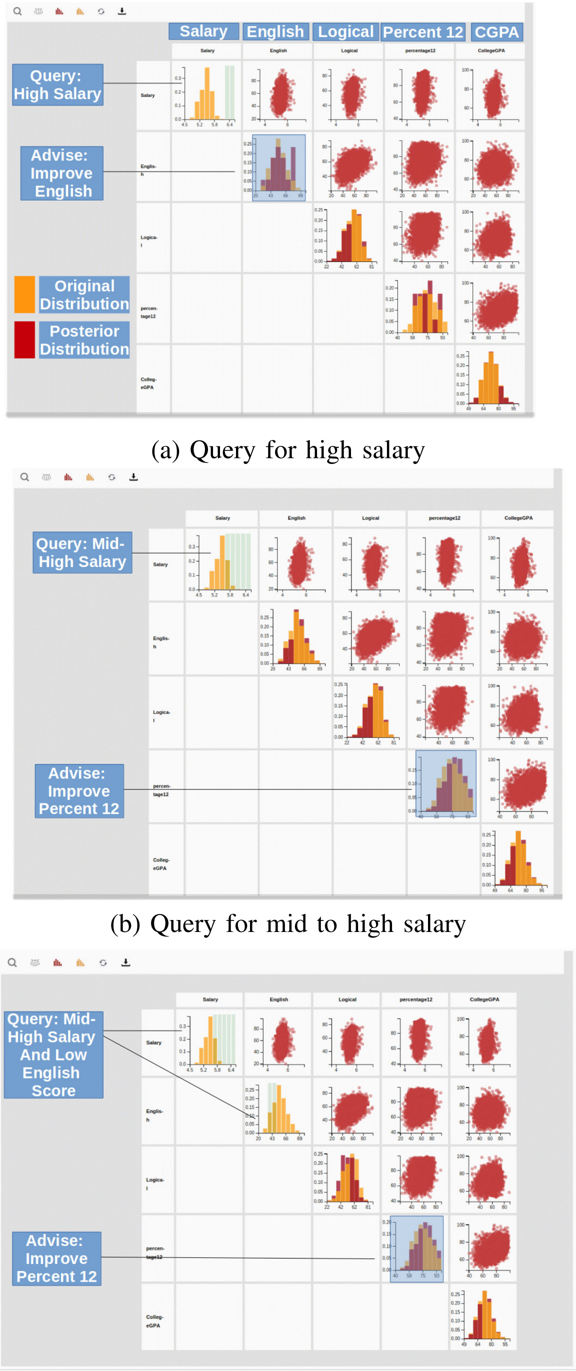
We consider the case when a candidate is interested in getting a very high salary and wants to know the ideal profile for the same. This may be done by selecting the last two bars on the salary distribution as shown in Figure 7 (a) and hitting the query button in the menu. It may be observed that there is a significant rise in the probability of the second last bin for English score, indicating that a candidate must have high English score to get a high salary. Additionally, we find that although the distributions of logical test score and CGPA do not change much, probability of 12th percentage increases significantly for the higher range bins. Thus, for a very high salary English score and 12th percentage must be high.

Next, we consider the case when a candidate is willing to lower the salary expectation to mid to high range, Figure 7 (b). In this case the distribution of English score changes only slightly for the higher bins, while for 12th percentage, probabilities for the mid to high bins increase significantly.

Finally, we consider the case when a candidate is interested in a mid to high salary but has low English score. Such a query may be performed by selecting the appropriate bars on the distributions of both salary and English score as shown in Figure 7 (c). This causes the distribution of logical ability to shift to the middle range while the distribution of 12th percentage shifts significantly to the higher bins. Thus, for a high salary with low English score, one must have a good logical test score and very good 12th standard percentage.

In this manner a candidate may impose conditions on any number of the variables in the network and get answers to how his/her profile should change in order to meet the salary goal.

Predict Attributes: We may use the saved Bayesian network to predict the target variable for new datasets in which it may be missing using imputation: The expected value of the target variable is computed from the posterior distribution obtained after Bayesian inference and used as the predicted



(a) Query for high salary

(b) Query for mid to high salary

(c) Query for high salary and low English score

Fig. 7. Recommendation using Linked Query View

value.

Another task posed by the challenge was to predict salary for a set of candidates for whom it was not provided. We used the MSTN, learned on the relevant feature subset, to predict the salary of each candidate in the test dataset using rest of the features in a network as evidence. We use exact inference accelerated by an SQL engine (as explained in our previous work [1]) which internally performs query optimization which is analogous to poly-tree based exact inference.

iFuse provides a visual interface for model-based prediction using parallel coordinates. The user selects a network to be used for prediction via imputation and a dataset with the target variable missing. We use a horizontal parallel coordinates plot so as to differentiate it from the exploratory parallel coordinates visualization as well as to indicate a network structure which is usually drawn in a top-down order even though the edges have no directionality in this case. The value of the attribute to be imputed is 0 for all data points initially as shown in Figure 8 (a). Clicking the 'Impute' button fires the imputation module at the backend and lines for the imputed values are moved to their position along the axis, Figure 8 (b).

A visualization of the predicted salary on test data created from of the training data in 70-30(%) ratio is shown in Figure 8. The error in prediction can be visualised in (b) on the last two axes - actual salary and imputed salary. We observe that salary is highly skewed, e.g. only 1% is greater than 10L and a single model cannot handle the skew in the target variable. This is one possible reason for high RMSE. (Ensemble learning using multiple Bayesian networks trained on different salary segments are envisaged in future enhancements to the platform.)

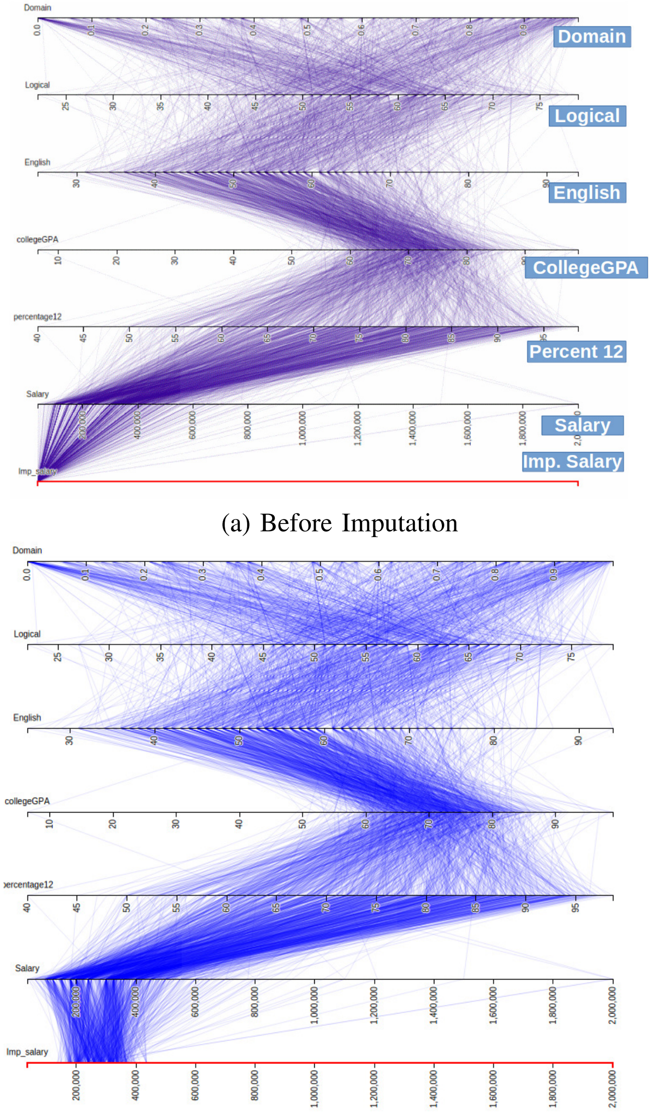
V. Product Data Lake

The third data lake (called 'product data lake' or PD-lake) consists of real-life data of consumer products from a global 'information and measurement' company. Available datasets in the product data lake are as follows:

1) **Local dataset**: contains set of items I = $\{I_r : r = 1, 2, ..., m\}$. For each item $I_r \in I$, we have local attributes, retailer descriptions, as well as measures such as sales figures.
2) **Global dataset**: contains the global market share of each item, where items are similarly described by global attributes.

Note that local attributes may be different from those used in the global dataset: these may include attributes not used at the global level. Even for similar attributes, local attributes use geography specific nomenclature, differing from global naming conventions. For example, in case of carbonated drinks, both global and local datasets may contains brand, flavour etc., of a drink, however, the actual values used may differ: e.g., 'Coke' vs 'Coca-cola', or 'Sweetened', vs 'Contains sugar'.

Additionally, local attributes include 'retailer descriptions', which are fields with free form text describing the product. Such text often points clues towards the global attributes for the item.



(a) Before Imputation

(b) After Imputation.

Fig. 8. Parallel Coordinates Plot for Salary Prediction using Imputation on test data created from 30% training data.

Consider a scenario where we need to compare country wise sales vs global market share of each brand. Given two datasets in PD lake, we cannot perform this task, as there is no natural join key between these two datasets. However, if we can predict value of global attribute g $\forall I_r \in I$ using local attributes and retailer description, we can perform the required task. In this section, we present an approach to make probabilistic prediction of g for items in local dataset using iFuse platform. Further, we join local dataset with global dataset using g as a join key and call it as *Probabilistic Join*.

Let g be the global attribute with n possible states say g_i, where $i = 1, 2, ..., n$. In order to predict value of g $\forall I_r \in I$, we use two different models, a) Bayesian Model and b) Text Information Retrieval Model which uses local attributes and retailer description respectively. We also calculate *confidence* of our prediction in both models. Further, we do ensemble of these two models to get the better prediction accuracy.

Bayesian Model (BM): In BM, we use only local attributes to predict global attributes for each item. We use the same approach as explained in section IV (Labour market data lake) which is explained as follows:

For a global attribute g, we select top K local attributes based on mutual information with g. Further, we learn MSTN with global attribute g and selected top K local attributes. Using an exact inference feature of iFuse, we calculate probability $\{p_i : i = 1, 2..., n\}$ of each state g_i of global attribute g, with local attributes as evidence in the MSTN. Finally, for every item, we choose the state g_i which have maximum probability p_i.

Confidence in BM Model ($Conf_{BM}$): For each item, we also calculate the confidence of the prediction of BM model. Given a probability distribution $\{p_i : i = 1, 2..., n\}$ for a global attribute g predicted using BM model. Consider an ideal distribution which is defined as:

$$q_i = \begin{cases} 1 & p_i = max\{p_i : i = 1, 2, ..., n\} \\ 0 & otherwise \end{cases}$$

and the confidence is given as:

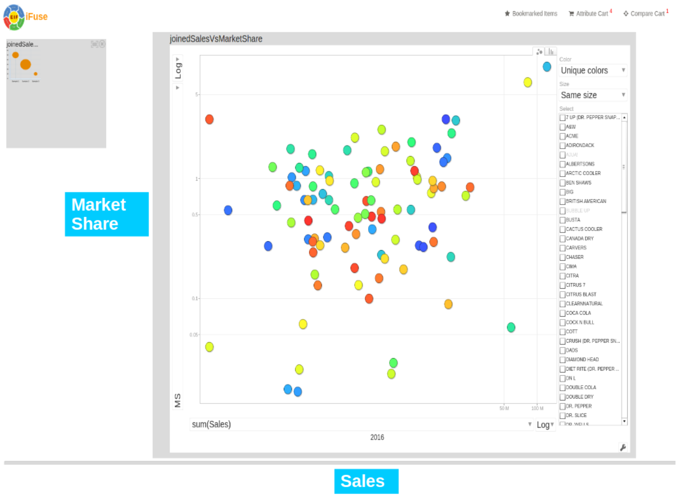
$$Conf_{BM} = 1 - \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \quad (2)$$

Text Information Retrieval Model (TIR): In TIR, we use retailer description to predict global attributes with the following approach

- Given the retailer descriptions of each item $I_r \in I$, concatenate all the retailer descriptions from all retailers into one string say S_r.
- Prepare a set of n-grams of adjacent words (up to bi-grams) in S_r. Let N be the set of n-grams n_k and let f_k be the frequency $\forall n_k \in N$.
- For each state g_i of g, we find best matching n-gram from the set N by calculating similarity score between g_i and every $n_k \in N$ using Jaro-Wrinkler algorithm and choose the n-gram which have max score. Let the score between g_i and the best matching n-gram is $s_{i,r}$.
- Multiply $s_{i,r}$ with the frequency of best matching n-gram to get new score for g_i, say $s'_{i,r}$.
- g_i with the maximum score $s'_{i,j}$ is our predicted value for global attribute g.

For each item I_r, we have two predictions (say g_B and g_T) of g using models BM (along with its confidence c_B) and TIR respectively. We do the ensemble of these two models using confidence values to get the better accuracy of prediction. For an item I_r, we choose g_B as the prediction of g if $c_B > t$, else we choose g_T as a prediction of g, where t is the threshold learned on validation set.

Once we predict global attribute of every item in a set I of local dataset. We join local dataset and global dataset after group by using g_i as a key in local datasets. Fig shows an example of joined dataset where we have data of carbonated drinks of one country. This visualization is called *Motion*



Color: Unique colors

Size: Same size

Fig. 9. Motion chart of datasets joined probabilistically

chart, it is an interactive visualizations for multi variate time-series data which can visually represent up to 4 temporal data attributes using position, size and colour of circles. In Fig 9, each circle shows the different brand and we have sales and market share of each brand mapped to x-axis and y-axis respectively. We choose unique colour for each brand and same size for each circle. This visualization helps to infer that some brands (top left corner) have high market share globally however, sales of those brands are quite low in particular country. Similarly, we can get interesting insights using various visualization in iFuse from datasets which lacks natural join key.

VI. Related Work

There are a number of tools that support visual analytics over data models created using machine learning techniques such as Spotfire [6], JMP [7] and Qlikview [8] of which only Spotfire provides Bayesian network models, learned using naive Bayes. In comparison *iFuse* supports naive Bayes as well as graph-based MSTNs. Additionally, we use the learnt model for filling missing data as well as a novel interactive visualization for querying the network, both of which are not available in Spotfire.

State of the art report on Linked views or multiple coordinated views is presented in [9]. As has been reported in this work, data pre-processing and preparation to support linked views at interactive rates is an important requirement. To the best of our knowledge, the only other work that uses Bayesian networks for linked views is our previous work [10]. In this work, we used a linked view consisting of a Heatmap and two Histogram plots. In *iFuse*, we have enhanced the design to show the histograms of attributes in the network as well as pair-wise scatter plots of the actual data in a grid layout.

Fusion of data from multiple sensors [11], [12], [13], [14], [15] has been attempted using models such as JDL[16]. However, this stream of work is focused towards object/target identification, threat assessment in defence arena, or in the context of robotics and machine intelligence[17]. However, to the best our efforts we could not find any work in the context of enterprise business intelligence with respect to fusion of data from disparate sources. We also augment such methods using our novel data visualization approaches.

VII. Conclusion

We have described the genesis of data-lake architecture for modern enterprise business intelligence, which offers significant flexibility as compared to traditional approaches. We have described a novel systematic procedure for navigating a data-lake, to derive insights, starting with searching and browsing of available datasets and joining of the chosen datasets. The joined datasets were then analyzed using Bayesian imputation for prediction of target variables, and such features are made available to analysts using novel data visualizations in our data-lake platform *iFuse*. All these features were described in the context of two data-challenges and a real-world enterprise data-lake. Finally, we placed our work in the context of related work in data visualization as well as in data fusion.

References

[1] S. Yadav, G. Shroff, E. Hassan, and P. Agarwal, “Business data fusion,” in *Information Fusion (FUSION), 2015 18th International Conference on*. IEEE, 2015.

[2] J. Zobel and A. Moffat, “Inverted files for text search engines,” *ACM Comput. Surv.*, 2006.

[3] J. Heinrich and D. Weiskopf, “State of the art of parallel coordinates.”

[4] G. Ver Steeg, “Non-parametric entropy estimation toolbox (npeet),” 2000.

[5] A. Kraskov, H. Stögbauer, and P. Grassberger, “Estimating mutual information,” *Physical review E*, vol. 69, no. 6, p. 066138, 2004.

[6] C. Ahlberg, “Spotfire: an information exploration environment,” *ACM SIGMOD Record*, vol. 25, no. 4, pp. 25–29, 1996.

[7] B. Jones and J. Sall, “Jmp statistical discovery software,” *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 3, no. 3, pp. 188–194, 2011.

[8] “Qlikview,” http://www.qlikview.com.

[9] J. C. Roberts, “State of the art: Coordinated & multiple views in exploratory visualization,” in *Coordinated and Multiple Views in Exploratory Visualization, 2007. CMV’07. Fifth International Conference on*. IEEE, 2007, pp. 61–71.

[10] G. Sharma, G. Shroff, A. Pandey, B. Singh, G. Sehgal, K. Paneri, and P. Agarwal, “Multi-sensor visual analytics supported by machine-learning models,” in *ICDM Workshop on Data Analytics meets Visual Analytics*, 2015.

[11] D. L. Hall and J. Llinas, “An introduction to multisensor data fusion,” *Proceedings of the IEEE*, vol. 85, no. 1, pp. 6–23, 1997.

[12] E. Waltz, J. Llinas *et al.*, *Multisensor data fusion*. Artech house Boston, 1990, vol. 685.

[13] M. Mutlu, S. C. Popescu, C. Stripling, and T. Spencer, “Mapping surface fuel models using lidar and multispectral data fusion for fire behavior,” *Remote Sensing of Environment*, vol. 112, no. 1, pp. 274–285, 2008.

[14] D. M. Buede and P. Girardi, “A target identification comparison of bayesian and dempster-shafer multisensor fusion,” *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 27, no. 5, pp. 569–577, 1997.

[15] P. Pinheiro and P. Lima, “Bayesian sensor fusion for cooperative object localization and world modeling,” in *Proc. 8th Conference on Intelligent Autonomous Systems*. Citeseer, 2004.

[16] A. N. Steinberg, C. L. Bowman, and F. E. White, “Revisions to the jdl data fusion model,” in *AeroSense’99*. International Society for Optics and Photonics, 1999, pp. 430–441.

[17] M. A. Abidi and R. C. Gonzalez, *Data fusion in robotics and machine intelligence*. Academic Press Professional, Inc., 1992.