CS 520 - Data Integration, Warehousing and Provenance

Paper Title - Explaining Dataset Changes for Semantic Data Versioning with Explain-Da-V

Group 12 - Fateen | Sahil | Himanshu

Topics to be Covered

Overview

High Level Summary of Explain Da-V

Methodology

Key Concepts and Methodologies of Paper

Analysis

Impact and Assessment of Explain Da-V

Introduction to Explain-Da-V and the Need for Semantic Data Versioning

1. ***Why Explain-Da-V?***

   - In projects where many people work on data, different versions of the same data are often created.
   - Traditional tools don’t explain well how and why data changes from one version to another.

2. ***What Does Explain-Da-V Do?***

   - It's a new tool designed to make sense of these changes.
   - Uses special methods to show how data has transformed between versions.

3. ***The Problem with Current Methods***

   - Usually, changes in data aren’t well recorded – like, who did what and why.
   - Explain-Da-V helps to clear up these mysteries in a simple way.

Introduction to Explain-Da-V and the Need for Semantic Data Versioning

4. ***Kinds of Changes Explain-Da-V Looks At:***

- It can understand changes in columns (vertical) and rows (horizontal) of data.
- Works with different types of data, like words, numbers, or categories.

5. ***What Makes Explain-Da-V Special?***

- It defines a new way to look at how data evolves over time.
- Sets up new standards for checking if the explanations it gives make sense.
- Proven to be better than older methods in tests.

6. ***Focusing on What's Inside the Data:***

- Mainly looks at changes within the data itself, like adding or removing information.
- Planning to explore more complex changes in the future.

Figure 1

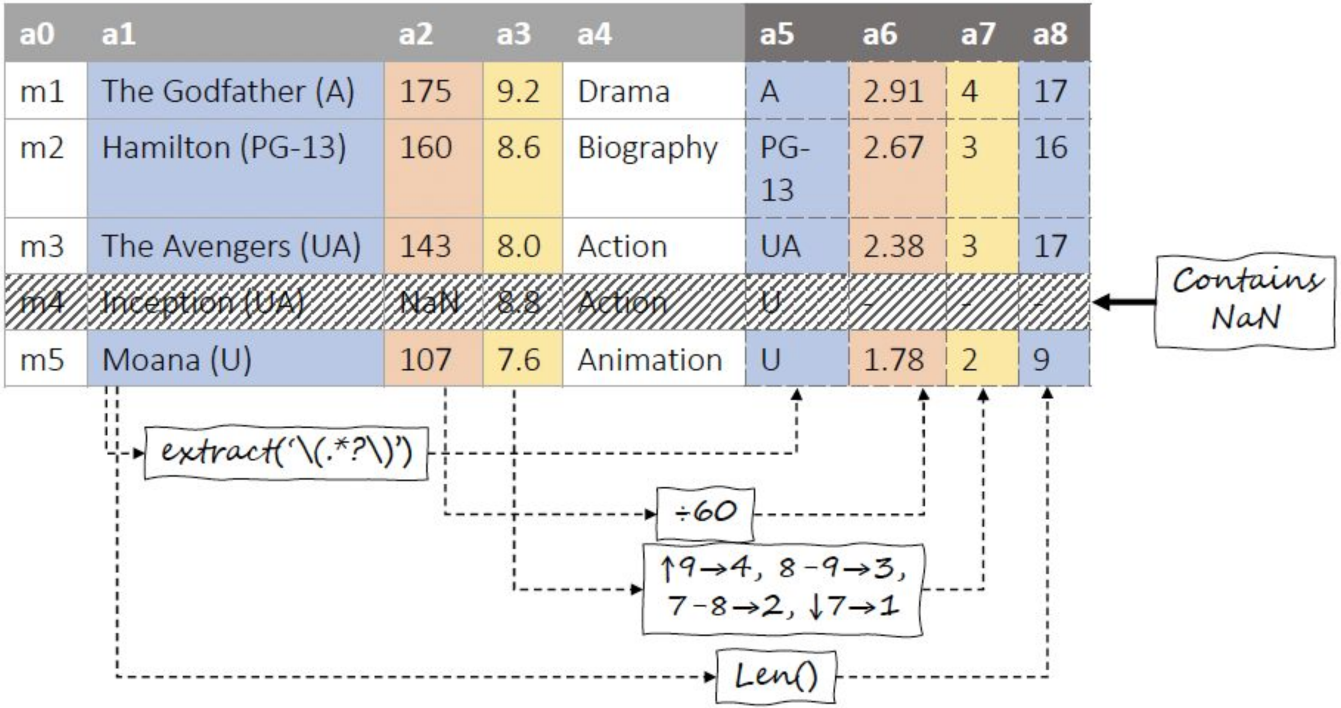
a0	a1	a2	a3	a4
m1	The Godfather (A)	175	9.2	Drama
m2	Hamilton (PG-13)	160	8.6	Drama
m3	The Avengers (UA)	143	8.0	Action
m4	Inception (UA)	NaN	8.8	Action
m5	Moana (U)	107	7.6	Animation

(a) Dataset version created by UserA

a0	a1	a2	a3	a4	a5	a6	a7	a8
m1	The Godfather (A)	175	9.2	Drama	A	2.91	4	17
m2	Hamilton (PG-13)	160	8.6	Drama	PG-13	2.67	3	16
m3	The Avengers (UA)	143	8.0	Action	UA	2.38	3	17
m5	Moana (U)	107	7.6	Animation	U	1.78	2	9

(b) Dataset version created by UserB

Figure 1-Changes Explained



a0	a1	a2	a3	a4	a5	a6	a7	a8
m1	The Godfather (A)	175	9.2	Drama	A	2.91	4	17
m2	Hamilton (PG-13)	160	8.6	Biography	PG-13	2.67	3	16
m3	The Avengers (UA)	143	8.0	Action	UA	2.38	3	17
~~m4~~	~~Inception (UA)~~	~~NaN~~	~~8.8~~	~~Action~~	~~U~~	~~-~~	~~-~~	~~-~~
m5	Moana (U)	107	7.6	Animation	U	1.78	2	9

Related Work - Exploring Data Versioning and Change

1. ***Data Versioning Research:***
   - Focus on developing version managers to manage and store different dataset versions efficiently.
   - Example tools: DataHub (git-like interface for datasets), TardisDB (SQL extension for version management).

2. ***Semantic Data Versioning:***
   - The focus is on understanding the semantic differences between dataset versions, not just version management.
   - Addresses gaps in schema versioning, assuming incomplete or ambiguous metadata.

3. ***Data Change and Integration:***
   - Based on principles of data integration, like attribute and tuple matching.
   - Past research includes change detection in structured and semi-structured data, and tools for exploring data/schema change (e.g., DBChEx).

4. ***Given Approach vs. Traditional Exploration:***
   - Unlike traditional focus on exploring 'what' and 'how many' changes occurred, we focus on explaining 'how' changes were made between versions.

Related Work - Data Transformation by Example

1. ***Programming-by-Example (PBE):***
   - Traditional approach: synthesizing programs from given input-output examples using various search spaces and algorithms.
   - Tools like Foofah and Clx use heuristic search to find transformations.

2. ***Transformation Repositories:***
   - Alternative methods involve creating transformation repositories from external sources like Web Forms, GitHub, Stackoverflow.
   - Tools like Transform Data by Example (TDE) and DataXFormer search for relevant functions from these repositories.

3. ***Data Preparation and Analysis Transformations:***
   - Research in data preparation transformations, focusing on tools like AutoPandas and Auto-pipeline.
   - Beyond the 'by-example' paradigm to include reshaping operations (e.g., group by).

Semantic Data Versioning

1. ***Defining Semantic Data Versioning:***
   - It's a systematic approach to track and interpret changes between different versions of a dataset.
   - Focuses on the meaning behind data alterations, not just the storage of multiple versions.

2. ***Alignment and Notation:***
   - Semantic versioning aligns corresponding elements between dataset versions to highlight changes.
   - Uses a notation system (L, R, V, A) to classify changes as matched (V) or unmatched (A), and whether they pertain to the original (L) or revised dataset (R).

3. ***Change Set Identification:***
   - Identifies added or removed attributes (vertical changes) and tuples (horizontal changes) between dataset versions.
   - Utilizes these identified changes to construct explanations for the transformations.

Figure 2

	Notation	Meaning	Notation	Meaning
basic	T	Left-hand dataset	T'	Right-hand (revised) dataset
	T_A	The attribute set of dataset T	T_r	The tuple set of dataset T
changes	$L\Delta_A$	Unmatched attributes in T	$L\nabla_A$	Matched attributes in T
		$\{A_i : A_i \in T_A \wedge \nexists A'_j \in T' : (A_i, A'_j) \in \Sigma_A\}$		$T_A \backslash L\Delta_A$
	$L\Delta_r$	Unmatched tuples in T	$L\nabla_r$	Matched tuples in T
		$\{\pi_{L\nabla_A}[r_j] : r_j \in T_r \wedge \nexists r'_i \in T : r_{0i} = r'_{0i}\}$		$\{\pi_{L\nabla_A}[r_j] : r_j \in T_r\} \backslash L\Delta_r$

Methodology for Explaining Dataset Changes

1. ***Change Explanations:***
   - An 'explanation' is defined as the set of transformations that convert the original dataset (O) to the revised dataset (G).
   - This transformation process provides a comprehensible narrative of the dataset's evolution.

2. ***Types of Data Changes:***
   - Vertical explanations address attribute-level changes while horizontal explanations deal with tuple-level changes.
   - The method differentiates between adding new information (additions) and removing existing data (removals).

3. ***Implementing Explain-Da-V:***
   - The framework systematically applies its core methods to elucidate vertical and horizontal changes.
   - It's designed to make sense of complex data transformations, enhancing transparency in data evolution.

Core Semantic Explanation Methods - Numeric Change

Limitations of linear functions: The linear functions are unable to cover the numerical transformations appropriately.

Expansion of feature space: The feature space (origin) gets expanded to generate additional features for enabling versatile transformations.

Polynomial Regression: Polynomial features are added to the origin to enable transformations that are polynomial in nature.

Inter-relation Features: Multiplying and dividing different attribute values in the origin help to enhance the transformation capabilities. These can be applied directly on the tuple level.

Extensions consecutively: Sequential extensions are applied to enable complex attributes that are difficult to solve by hand (eg. Applying the BMI formula).

Mathematical transformations: The mathematical extension of the origin are used to cover common maths formula such as log, squareRoot, reciprocal and exponent.

Transformations of the tuple: The tuple-specific maths transformations are applied.

Global aggregators: Attributes like sum, mean, max, and min are generated to cover ML transformations like normalization. This value is compounded by the values in the attribute.

Regression fitting: Extended feature set fits on a regressor, to assign coefficients on the extended feature set.

Learning Transformation: The transformation gets fitted on T and T' without training data

Core Semantic Explanation Methods - Categorical Change

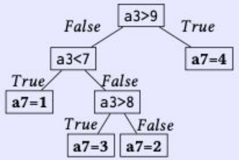
1. ***Problem framing:*** The problem is defined as a classification task to define categorical goal using an origin relation of numeric data.

2. ***Classification Framework:*** Explanatory variables are listed as the origin relation's tuples. The goal serves as the output class labels.

3. ***Binary/Multi-class output:*** Output can be binary (movie is longer than 2 hours or not) or multi-class (grade of a student).

4. ***Explainability focus:*** uses decision trees to understand the problem statement. This is because decision trees can represent the explanations in disjunctions or conjunctions. The path from root to lead is a conjunction and the tree is a disjunction. They provide a combination of conditions that are interpretable and help understand the categorical goal of the numeric origin.

EXAMPLE 7. *Figure 1b provides an example a categorical transformation, namely,* a7, *which can be resolved with the help of the following decision tree.*



Core Semantic Explanation Methods - Textual Change

1. ***The PBE approach for the textual origin:***

When we have to deal with text based data, we employ the PBE (Programming by Example) approach. Eg, Foofah, an existing framework.

2. ***Search – based Solution:***

The search algorithm generated by the PBE explores operator space using a heuristic function to uncover cost of proposed solution. Here we can also note that pruning the space enhances search speed.

3. ***Textual-to-text transformations:***

Conventional NLP steps like lemmatization, removal of special characters etc can be handled by extending traditional PBE steps. The repository keeps a note of the implemented operators.

4. ***Text-to-numeric transformations:***

We use a meta-operator to count occurrences of a pattern pat in the value. The operations will be defined and we also count the pre-determined sets of stop-words.

5. ***Text-to-categorical transformations:***

A similar meta-operation for pattern existence can be defined and contains_percent = contains‘%' can be formed using this logic.

Core Semantic Explanation Methods - Categorical Encoding Change

1. ***Handling mixed data types:***

We use one-hot-encoding to deal with mixed data types, specifically textual and categorical based.

2. ***Values assignment:***

We assign 1 to specific categorical value and 0 to others to make everything uniform.

3. ***Ordinal encoding:***

We use this encoding technique to enhance the transformations and provide more intricate representations.

Example:

We can use this encoding technique to classify prediction of movie ration (eg. "Is_Drama" or "Is_Action") to enhance the learning process.

Core Semantic Explanation - Finding the Origin

1. ***Search Optimization:***
   - Introducing the concept of narrowing the search to a subset of the original dataset 'T' to improve efficiency.
2. ***Identifying the Origin:***
   - The challenge lies in identifying the subset of data (origin) used to derive the specific goal (new version).
3. ***Problems with Using Full Dataset:***
   - Using all attributes of 'T' as the origin may introduce noise, complicating the search with irrelevant data.
4. ***Functional Dependencies:***
   - Creating a new attribute inherently establishes a functional dependency between the new and original attributes.
5. ***Using Functional Dependency Algorithms:***
   - Deployment of algorithms to detect these dependencies, which helps to determine the original subset that explains the new attribute.
6. ***Multiple Potential Origins:***
   - Recognizing that multiple attribute sets could define the goal, leading to the identification of several potential origins.
7. ***Selecting Among Multiple Origins:***
   - Analyzing various origins by examining attribute sets, considering their size and distinct values (cardinality).
8. ***Early-Stop Condition:***
   - Implementing an early-stop condition based on the size or transformation quality to streamline the search process.

Explaining Vertical Changes in Datasets

1. ***Attribute Additions:***

   - Newly added attributes are usually the result of applying transformations to existing data, often for feature engineering in machine learning.
   - Explain-Da-V identifies the most relevant original data (the origin) and then applies explanation methods to elucidate the transformation to the new attribute.

2. ***Iterative Explanation Process:***

   - The process tackles one added attribute at a time, searching for its possible origins and then applying case-based explanation methods depending on the data type involved.

3. ***Handling Attribute Removal:***

   - Attribute removals are treated as individual cases and often reflect data cleaning steps like removing duplicates or attributes with excessive missing values.
   - The system determines whether an attribute was removed based on set thresholds, such as a high ratio of missing values, or due to duplication of information.

User c made changes to figure 1a

a0	a1	a2	a3	a4	a9	a10	a11	a12	a13	a14
m1	The Godfather (A)	175	9.2	Drama	0.28	3.15	1	godfather a	8.9	1
m2	Hamilton (PG-13)	160	8.6	Drama	0.28	3.23	1	hamilton pg	8.9	1
m3	The Avengers (UA)	143	8.0	Action	0.24	3.36	0	avengers ua	8.0	2
m5	Moana (U)	107	7.6	Animation	0.23	4.26	0	moana u	7.6	3

Explaining Horizontal Changes in Datasets

1. ***Tuple Removal:***
   - Removing tuples is common in data cleaning, which involves eliminating data points based on specific criteria.
   - Explain-Da-V examines each removed tuple individually to provide explanations, which may involve identifying and removing duplicates or outliers.

2. ***Collective Tuple Removal:***
   - In some cases, tuples are removed en masse due to missing values or other collective criteria.
   - The method seeks to explain these group removals through common patterns or categorical methods, particularly when dealing with mixed-type data.

3. ***Tuple Addition:***
   - Non-idiopathic tuple additions, such as those resulting from oversampling, are explored to identify if they've been duplicated from existing tuples.
   - This step helps in recognizing bootstrapping operations and other data augmentation processes.

Assessing the Explanation's Quality

1. ***Explanation Evaluation:***
   - Focus on generating explanations that not only replicate the changes but also generalize beyond specific instances.
   - Multiple valid explanations may exist for a single change; the challenge lies in selecting the most accurate one.

2. ***Explanation Validity:***
   - Validity measures whether the transformation accurately recreates the goal from the origin.
   - It is quantified as a success rate based on the proportion of correctly transformed tuples.

3. ***Generalizability Assessment:***
   - Goes beyond validity by measuring how well an explanation applies to similar data versions.
   - Ensures the solution's effectiveness across different datasets within a data pipeline.

4. ***Choosing the Right Explanation:***
   - Among multiple valid explanations, preference is given to those with higher validity.
   - Explainability dimensions, such as conciseness and concentration, play a crucial role in selecting the best explanation.

Measuring Explainability and Generalizability

1. ***Explainability Dimensions:***
   - Conciseness: The fewer the components in a model, the more understandable it is. This includes the number of coefficients in regressions or nodes in decision trees.
   - Concentration: Focused explanations with fewer chunks of information are preferable, aligning with human working memory limits.

2. ***Total Explainability:***
   - A combination of conciseness and concentration, weighted according to user or system preference.

3. ***Early-Stop Condition in Searches:***
   - To manage large search spaces, explanations are sorted by the **size** and **distinctness** of their origin.
   - Search stops early if an explanation meets a predefined threshold of validity and explainability, enhancing efficiency.

4. ***Generalizability in Practice:***
   - Can only be used for selecting explanations if additional dataset versions are available for comparison.
   - Vital for applications in ETL processes, where changes are consistent across datasets.

Evaluating Explain Da-V's Performance

1. ***Benchmarking Approach***:
   - Established a new benchmark, Semantic Data Versioning Benchmark (SDVB), with **342** dataset versions covering various data transformation scenarios.
   - Adopted an existing dataset from Yang et al. to evaluate the synthesis of data pipelines.

2. ***Performance Indicators***:
   - Validity: Measures if the transformation precisely re-creates the goal from the origin.
   - Generalizability: Assesses if the transformation applies to similar dataset versions in different contexts.

3. ***Comparison with Baselines:***
   - Explain-Da-V outperforms other methods, showing particular strength in handling diverse data types.
   - In scenarios involving numeric data, Explain-Da-V displays a significant performance advantage.

Insights from Explain Da-V Ablation Study

1. ***Ablation Study Components:***

- Analyzed the impact of finding the origin and the inclusion of numeric-to-numeric transformation extensions on the performance.
- W/O find origin and W/O extensions
- Assessed how treating all attributes as either numeric or textual affects results.

2. ***Key Findings:***

-The full implementation of Explain-Da-V yields the most valid and generalizable explanations.

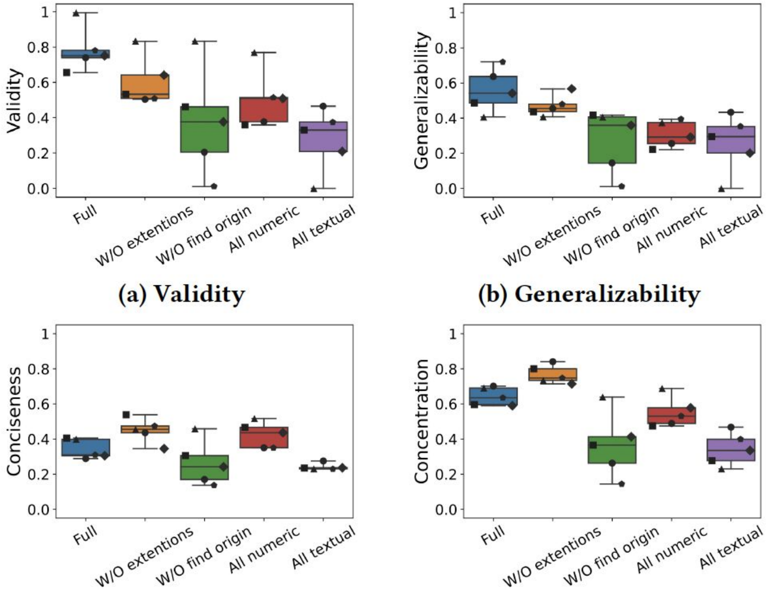
-Finding the origin and adding extensions significantly increases validity by **30%** and **107%** respectively..

- Treating attributes as numeric or textual without considering their **true type leads** to decreases in validity by 35% and 64%, respectively.

3. ***Explainability of Transformations:***

- While transformations without extensions are more straightforward, they may lack in validity and generalizability.
- Numeric transformations tend to be more concise and easier to comprehend compared to textual explanations.

Dataset→ ↓Method	IMDB			NBA			WINE			IRIS			TITANIC			Auto-pipeline		
	Val	Gen	# $\mathcal{E}$	Val	Gen	# $\mathcal{E}$	Val	Gen	# $\mathcal{E}$	Val	Gen	# $\mathcal{E}$	Val	Gen	# $\mathcal{E}$	Val	Gen	# $\mathcal{E}$
Foofah	.42	.42	3.7	.28	.28	4.2	.29	.29	3.9	.23	.23	3.1	.29	.29	4.1	.55	-	3.3
Foofah+	.44	.44	3.7	.29	.29	4.2	.34	.34	3.9	.25	.25	3.1	.37	.37	4.1	.55	-	3.3
Auto-pipeline*	.44	.44	3.7	.30	.30	4.2	.33	.33	3.9	.26	.26	3.1	.37	.37	4.1	.78	-	3.3
Explain-Da-V	**.73** (.64)	**.60** (.56)	6.4	**.90** (.89)	**.79** (.69)	7.3	**.87** (.76)	**.81** (.59)	6.8	**.93** (.88)	**.83** (.76)	8.9	**.88** (.79)	**.77** (.68)	7.2	**.82** (.78)	-	5.7
+ over baseline	+65%	+36%		+202%	+167%		+156%	+138%		+254%	+217%		+140%	+109%		+5%	-	



(a) Validity

(b) Generalizability

Figure 7: Ablation Study over SDVB datasets, namely, IMDB (■), IRIS (▲), WINE (•), NBA (⬣), TITANIC (♦)

Explainability Components

-While explanations without extensions are more concise and concentrated, they tend to be less valid and generalizable.

- When the origin is not found, explanations are less concise and less concentrated.

-Numeric explanations are shown to be more explainable than textual explanations, particularly in terms of conciseness.

Version-Sets Performance

- The NBA version-set exhibits **low validity** and **generalizability** without finding the origin, highlighting the importance of this feature for datasets with diverse attributes.
- The IRIS version-set, mainly consisting of numeric attributes, shows very low performance when all attributes are **treated as textual**, indicating the necessity of accurately identifying attribute types.

Conclusion

The work established a foundation for explaining semantic changes in data versioning and demonstrated the effectiveness of Explain-Da-V against multiple baselines.

We realised that the performance of Explain-Da-V on numeric-to-numeric conversion was the highest.

However factors such as numeric-textual and treating attributes regardless of their type brought in performance decline.

Verdict- The paper explain sufficiently how well the version changes are explained by this tool, however it needs to perform better in the bad conditions too thereby making it a one stop solution.

Thank you