Aikido: Structuring data point identifiers of technical building equipment by machine learning

Florian Stinner¹, Paul Neißer-Deiters¹, Marc Baranski¹ and Dirk Müller¹

¹ RWTH Aachen University, E.ON Energy Research Center, Institute for Energy Efficient Buildings and Indoor Climate, Mathieustraße 10, 52074 Aachen, Germany

E-mail: fstinner@eonerc.rwth-aachen.de

Abstract. Building automation and control systems (BACS) generate large amounts of data. Manual interpretation of the data, e.g. for identifying energy efficiency measures, is in many cases either too costly or too complex. In order to reduce the manual effort of building data analysis, it is necessary that the labeling of data points is human- and machine-readable. As there is currently no widely accepted standard for labeling, these labeling texts differ greatly. We developed an algorithm based on Natural Language Processing (NLP) that automates the translation of BACS metadata. In previous works, we developed a schema based on 40 other schemas for the designation of data points (BUDO Schema) that we use for the translation. The user labels data points suggested by our algorithm Aikido and checks the prediction in a browser-based GUI (key interface). In our case study, we use a dataset that comprises five different projects, each using its own schema for structuring the metadata of data points in BACS. After ten training labels, our algorithm achieved an accuracy of 0.87 across all data sets. This increased to 0.95 with 50 training labels. We show that the chosen approach is able to normalize the metadata of different schemas to transfer to a previously defined schema. Thus, the manual effort is decreased significantly.

1. Introduction

Building Automation and Control Systems (BACS) generate big data volumes. For the building operators, it is difficult to analyse all data. Therefore, many energy saving potentials remain unused for cost reasons. Automated analysis and control functions can support the building operator in optimizing the operation. However, the system must also be machine-readable for this purpose. Data points in BACS are have certain labels. The quality of the metadata varies depending on the person who created the labels.

There is no industry-wide accepted standard for structuring data point (DP) labels that covers nearly all required labels of the building automation practice ([1] [2]). This results in a lock-in effect as building operators are reliant on BACS using the standard installed of their building automation vendor. Only the building automation vendor can process the installed standard. Efforts in the past were made to introduce an industry-wide standardized format, as it would reduce the cost of installing new BACS significantly.

The most used protocol for BACS is BACnet [3]. A new VDI guideline was released in January 2019, which defines a text identifier of fixed length containing only German abbreviations [4]. Knowledge-based representations are Project Haystack [5] or Brick Schema [2]. Acceptance and
market penetration of any of these standards is based on its application in new and existing BACS. In our approach, we use the BUDO Schema, which is based on 40 standards provided by different operators of BACS [6]. It combines the advantages of a text identifier with fixed length with the advantages of knowledge-based representation. It can be used in BACnet and thus in the state of the art.

The normalization of metadata to a common metadata schema is essentially for automated analysis and control functions. Manually executed, transformations are mostly not cost-efficient. Automated algorithms could support the transformations. For real application, it is crucial that the human effort is reduced.

[7] and [8] evaluated different algorithms with the data of various buildings. Previous approaches for the normalization of metadata focused on reducing effort, but not on the application by people with little knowledge of computer science. The approach presented in [9] shows the best results. In this approach, DP designations are used across buildings to decode them. The meaning of each letter of DPs label must be defined exactly. However, the user is not guided through the definition, but has to know the representation of each category.

In this paper, we avoid the problem that users with knowledge in building operation must also have computer science knowledge. We use an approach from active learning, where DPs are suggested to the user that are most similar to the most frequently occurring unclassified DPs.

We structure the paper as follows: first, we describe the method with the used data sets, the used BUDO Schema and the developed algorithm. Afterwards, we show the results and discuss further improvements.

2. Method

2.1. Used data sets

Our algorithm is tested with a large number of data sets containing different labeling formats. This is important, as its usage is only reasonable if it achieves sufficient reduction of human effort in most use cases. Table 1 shows the variety of the used DP labeling. In this paper, we used five data sets, each including more than 100 datapoints.

Table 1. Used data sets of data point labels. Each including more than 100 datapoints.

| Number | Number of data points | Example |
|--------|------------------------|---------|
| (1)    | 482                    | G03_0.PL-WP2_WMZ_Trl |
| (3)    | 253                    | FFB_B1.C15_AIR_IWT1.1_T.15.8.8 |
| (6)    | 223                    | BKT_Temperatur_VL_GH |
| (7)    | 392                    | ISP01_AS01/RLT Küche/Lüftung Küche Ablufttemperatur |
| (8)    | 221                    | 4120.MONI.AEMW141 RR1026 FVU T h out |

2.2. BUDO Schema

The BUDO Schema we use in this paper defines two types of categories, which are represented in a label. Categories can be divided into two types. A category that is not further identifiable, here referred to as Type I category, is, for example, a room number. Type II categories are further classified with a given vocabulary of abbreviations. An example is the Type II category medium. A medium could further be identified as water or air. This classification can be detailed by adding up to three additional levels. An additional level for water could be drinkable. The format allows for a detailed identification if needed, but also allows a short label. For simple DPs, all categories are optional [6].
2.3. “Artificial Intelligence based Key Interface for Data point metadata nOrmalisation” (Aikido)

The “Artificial Intelligence based Key Interface for Data point metadata nOrmalisation” (Aikido) is a User interface to structure existing data point labels with the help of an algorithm. Suggested by a selection algorithm, the user must correctly assign a certain number of DPs. With the help of a two-layered prediction algorithm (sequence labeling, multilabel classification), the assigned DPs were used to predict the further DPs of the data set. The labeling process is repeated until the user has marked all DP labels as correct.

In the first layer, the expected positions of the categories of the DP are detected and assigned. In the second layer, if possible, the categories are specified more precisely. For example, the category system is more exactly classified with boiler. In each step, we show the results of suitable algorithms mostly based on artificial intelligence (AI).

![Figure 1. Example for the entire two-layered algorithm with sequence labeling and classification.](image)

2.3.1. User interface

The Aikido tool is a browser-based application utilizing the Django framework. Django allows a quick development of a web interface using a data base to handle its content. The user can upload DPs as CSV file. Figure 2 shows the uploaded data sets and figure 3 shows the included original labels of one data set. The data sets and the categorized DPs are saved in a SQL database. The Aikido algorithm selects the most suitable label for the algorithm and the user has to define all corresponding parts in the user interface. Predicted DPs can be verified and, if necessary, corrected by the user and used to optimize further prediction for higher accuracy for the following predictions.

2.3.2. Selection of suitable data point labels

The information gain per DP label has to be maximized to achieve a steep learning curve. Data points are rated higher if they are similar to yet unclassified DPs and different to already classified DPs. The highest rated DP is selected as the most suitable. Similarity and difference are calculated by the Levenshtein metric. This algorithm is called Levenshtein selection.

2.3.3. Sequence labeling

The first layer applies sequence labeling algorithms to extract substrings from the initial label and assign them to the BUDO Schema categories. As the labeling consists of natural as well as formal language, a word-based approach is not applicable.
Therefore, it is necessary to separately assign each character in a label to a category. Characters assigned to the same category are joined as a substring within the initial label.

The features observed for the sequence labeling are the ASCII representation of the six nearest neighbours, as well as information about the current character and the number of separating characters, e.g. ‘.’ or ‘ ’, before and after the currently observed character.

We tested four different sequence labeling algorithms, two conditional random field (CRF) algorithms, a hidden markov model (HMM) and a structured perceptron algorithm. These algorithms have been successfully used in Natural Language Processing (NLP) and DP labels are structured in an artificial ‘language’.

### 2.3.4. Multilabel classification
Categories within the BUDO Schema are classified by multilabel classification algorithms. We tested three different multilabel classification algorithms, a k-nearest neighbour algorithm (KNN), a random forest classifier (RFC) and a multilayer perceptron (MLP). We used a k of 3 for KNN. RFC is used with 10 trees in the forest. For MLP, we used logistic sigmoid as activation function. The algorithms are supplied with ASCII representations of a substring.

### 3. Results
As the reduction of manual effort is the focus of this publication, a performance indicator is plotted in each figure as a function of the number of training DPs. We have used the other DPs of the datasets as test data (see Table 1). If possible, we used the f1-score, as it is a better indicator for performance on unbalanced data sets than accuracy. However, for results that could not be identified as true or false, accuracy was measured as f1-score would not correctly represent the results.

Figure 4 shows the f1-scores achieved by the sequence labeling layer. Data points are considered as correctly predicted if the exact substring is extracted. The CRF-algorithms performed best with the wapiti CRF algorithm achieving a higher f1-score than the sklearn CRFSuite CRF algorithm. At 50 training DPs, it reaches an f1-score of 0.95. As it outperforms all other sequence labeling algorithms, we select it for the sequence labeling layer.

Figure 5 shows the performance of the multilabel classification algorithms under the assumption of a correct sequence labeling layer. The sklearn RFC algorithm performs slightly
better than or equal as the other algorithms over all numbers of training DP labels and reaches an accuracy of 0.93 at 50 training DP labels. It is therefore selected for the Aikido algorithm. For a low number of training DPs, the Levenshtein selection performs, on average, slightly better than a random DP selection, as figure 6 shows. For higher numbers of training DP labels, the random selection outperforms the Levenshtein selection. This indicates that the difference to already classified DPs is not weighed high enough in the rating.

Figure 7 shows the overall performance of the Aikido algorithm for different buildings. The performance depends heavily on the data set. For building 1 and building 8 an accuracy over 0.97 was reached, while building 6 only reached an accuracy of 0.82 at 100 training DP labels. However, for all data sets, even with only one training sample, accuracies above 0.6 are reached. Therefore, the necessary user inputs are reduced significantly for every tested data set.

4. Discussion
On the basis of fewer training DPs, our algorithm could assign the further DPs of the data set correctly. The idea that the categorization of the individual labeling schemas of buildings are similar can be confirmed on the basis of the available results for this data set. One exception
(building 6), where several schemas were used, also produced significantly worse results. The only slightly better results of the Levenshtein selection compared to a random selection are astonishing. Currently, the Levenshtein selection is only used before the first step (sequence labeling). The application of a two-step procedure for the selection of DPs (sequence labeling and multilabel classification) may lead to better results. The selection of further distances such as Hamming or Jaro-Winkler distance could also improve the results.

Integrating expert knowledge is a high effort with every approach. We have been able to reduce the necessary expert knowledge, but it is still necessary to first have a technical specialist. Using the existing standards on which the BUDO Schema is based or using the previous labeled data sets would reduce the initial effort.

5. Conclusion
We were able to show that our developed algorithm (Aikido) is able to rename data points of different labeling systems. By using our user interface even users not familiar with computer science are able to execute the algorithm. The results are very promising with two datasets with an accuracy of 0.97 after 100 training DPs, that is nearly to the results presented in [7] and [8]. The results has to be compared and evaluated on the same database. The metadata database presented in [8] could be a good first step, but their development is not finished yet.

The presented approach enables the application of any labeling systems. In addition, advanced algorithms for the automated detection and implementation of energy saving potentials can be applied. Nevertheless, the algorithm can be further developed. The current speed of algorithm development in the Natural Language Processing (NLP) area is very high. These developments offer a high potential for improvement.

6. Acknowledgments
The authors would like to acknowledge the financial support of the German Federal Ministry of Economic Affairs and Energy (promotional reference 03SBE0006A).

References
[1] Stinner F, Yang Y, Schreiber T, Bode G, Baranski M and Müller D 2019 Generating generic data sets for machine learning applications in building services using standardized time series data ISARC - 36th International Symposium on Automation and Robotics in Construction
[2] Balaji B, Bhattacharya A, Fierro G, Gao J, Gluck J, Hong D, Johansen A, Koh J, Ploennigs J, Agarwal Y, Bergés M, Culler D, Gupta R K, Kjærgaard M B, Srivastava M and Whitehouse K 2018 Applied Energy 226 1273–1292
[3] Deutsches Institut für Normung 2017 Din en iso 16484-5:2017 - building automation and control systems (bacs) - part 5: Data communication protocol (iso 16484-5:2017);
[4] Verein Deutscher Ingenieure 2019 Vdi 3814 blatt 4.1 - Gebäudeautomation (ga) - methoden und arbeitsmittel für planung, ausführung und übergabe - kennzeichnung, adressierung und listen
[5] Project Haystack 2019 Project haystack URL https://project-haystack.org/
[6] Stinner F, Kornas A, Baranski M and Müller D 2018 Structuring building monitoring and automation system data The REHVA European HVAC Journal - August 2018 REHVA Journal ed REHVA pp 10–15
[7] Gao J and Bergés M 2018 Advanced Engineering Informatics 37 14-30
[8] Koh J, Hong D, Gupta R, Whitehouse K, Wang H and Agarwal Y November 7-8, 2018 Plaster: An integration, benchmark, and development framework for metadata normalization methods BuildSys ’18 ed ACM URL https://www.synergylabs.org/yuvraj/docs/kohBuildSys18_plaster.pdf
[9] Koh J, Balaji B, Sengupta D, McAuley J, Gupta R and Agarwal Y 2018 Scrabble: Transferrable semi-automated semantic metadata normalization using intermediate representation BuildSys’18 ed Ramachandran G S and Batra N (The Association for Computing Machinery) pp 11–20 ISBN 9781450359511