Towards Rational Feature Selection and Prioritization in Business Domain: A Big Data and Goal-Oriented Approach

Haan Mo Johng, Gonsoo Moon, and Seungtack Baek

Abstract—Machine learning techniques have been adopted for various business applications due to an ability to provide useful insights, learned from data collection. However, selecting meaningful data features is a difficult and time-consuming process because it is tricky to know which sets of data features are important and which combinations of data features need to be tested preferentially. In business, knowing the meaning and importance of the data is dependent on understanding a domain. Many excellent researches have done regarding feature selection, however, it seems that there is a room to study regarding understanding domain and feature selection. To approach this problem, goal-oriented feature selection and prioritization approach is proposed in this paper. This approach helps in selecting important data features and in prioritizing different combinations of data features to be tested preferentially. For validation, the approach is applied to the price prediction problem by using the Airbnb New York dataset with 36 data features and 40,587 instances. Gradient-boosted tree regression in Spark MLlib is used by changing different combinations of data features selected by the goal-oriented data feature selection approach we propose in this paper. The results are measured in RMSE and MAE, and compared with a result of using all data features. Maximum 14.10% of improvement of MAE and 18.46% improvement of RMSE are observed.

Index Terms—Machine Learning, Feature Selection, Feature Prioritization, Goal-Oriented, Goal-Orientation.

1 Introduction

Machine learning techniques have been adopted for various business applications due to the ability to provide meaningful insights about a task by learning from data collection. Since the meaningful insights can be obtained from meaningful sets of data features that are highly related to the task, selecting meaningful data features among all data features is important [1–4].

However, selecting meaningful data features is a difficult and time-consuming process because it is tricky to know which sets of data features are important and which combinations of data features need to be tested preferentially. Without understanding importance of each data feature, data features will be selected intuitively or randomly and different combination of data features will be tested every experiment until obtaining reasonable performance. This problem can be exaggerated when the number of data features is large.

In business, knowing the meaning and importance of the data is dependent on understanding a domain. Many excellent researches have done regarding feature selection [1–4], however, it seems a room to study regarding understanding domain and feature selection.

To approach this problem, we propose the goal-oriented feature selection and prioritization approach. Our approach helps in selecting important data features and in prioritizing different combinations of data features to be tested preferentially.

In our approach, we identify stakeholders related to the task, the stakeholders' goals, and relationships between the goals and data features. The stakeholders are classified in terms of importance degree, and data features, which are related to most important stakeholders goals, are considered as important features. Likewise, data features that are related to less important stakeholders' goals are considered as less important data features. This approach can be helpful to rationally infer important data features that might have positive impacts on a task by understanding a domain and then select important data features systematically.

After identifying degree of importance of data features, we construct combinations of the data features and prioritize the combinations of data features. The combinations of data features that include more important features would have higher priority. Prioritizing each combination of data features can be helpful in determining which combination of data features need to be tested preferentially.

We applied our approach to the price prediction problem by using the Airbnb New York dataset with a total of 36 data features and 40,587 instances. We used Gradient-boosted tree regression in Spark MLlib to predict the price and applied our approach for better feature selection. We measured the performance of our approach in terms of Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). We observed 14.10% of improvement of MAE and 18.46% improvement of RMSE.

2 Background

2.1 Goal-Oriented Requirements Engineering with the NFR Framework

The success of software project is dependent on how well people can deal with stakeholders, stakeholders' goals, and stakeholder requirements. The goal-oriented requirements engineering is to deal with identifying stakeholders' goals, exploring alternatives to achieve the goals, selecting among

ones, and prioritizing the goals and ones. Since most of stakeholders' goals are expressed in subjective with non-functional terms, which are called soft-goals [5, 6], NFR Framework has been proposed for dealing with the stakeholder softgoals [7].

The Softgoal Interdependency Graph (SIG), a part of the NFR framework, is useful for depicting, analyzing, and reasoning about the softgoals goals that are expressed not completely, but in a good enough sense. The SIG assists in reasoning about the interactions and conflicts among the stakeholders' goals and in showing a level of importance of a goal relative to other goals in the goal model.

We adopt the NFR framework and SIG to depict stakeholders, stakeholders' goals, and relationships between stakeholders' goals and data features for selecting and prioritizing data features in business domain.

2.2 Airbnb, Business Domain

We apply our approach to the Airbnb dataset. The Airbnb has emerged as an alternative supplier with great impacts on hotel business. Airbnb is a pioneer in shared accommodations that help people to rent their room to other people, and is considered as an alternative of lower-end hotels and hotels not catering to business travelers [8].

People renting their rooms by using Airbnb platform are called Airbnb hosts, and Airbnb hosts are responsible to decide the price of their room for themselves. However, since Airbnb hosts need to decide the price for themselves, it is hard for Airbnb hosts to know a proper price of their room.

2.3 Big Data

Since the data size and the data growth rate in industries is increasing incredibly, and types of data sources are getting diverse, big data technologies has been adopting broadly in industries to process, store, and analyze the big data in just time.

Spark, distributed and scalable computing platform to process and analyze the big data in a fast way using memory, has been adopting to industries widely. This is due to not only the functionalities of fast and scalable distributed computing, but also the functionalities of Machine Learning library for data analysis, SQL, Graph, and Streaming. We adopt the Spark to use the machine learning library, so-called Spark MLlib.

Hadoop [11] is also a good alternative for storing, processing, and analyzing big data, but it has lack of support for machine learning libraries. Flink [12] is also an good alternatives for big data processing with machine learning libraries, but it is more focus on streaming and provides less number of machine learning algorithms than Spark. Tensorflow [13] is a good alternative for fast machine learning processing using GPU and TPU, however, it has lack of support for distributed computing.

Particularly, we use Gradient-boosted tree regression in Spark MLlib to predict price of Airbnb listings for Aitbnb hosts. Spark MLlib provides Linear regression, Generalized linear regression, Decision tree regression, Random forest regression, Gradient-boosted tree regression, Survival regression, and Isotonic regression for regression problems [9]. In general, Gradient-boosted tree, an ensemble technique, shows better performance than linear regression, decision tree, and random forest regression, we choose the Gradient-boosted.

Spark documentation provides usage tips for tuning Gradient-boosted tree regression [10]. The document recommends to tune loss, numIterations, and algo parameters. To simplify our experiments, we use default value for loss and algo, and change numIterations from 10 to 50.

2.4 Machine Learning and Feature Selection

In general, feature selection is one of the most critical areas to build good machine learning applications. Selecting proper and relevant features to the task is essential to process to build and reliable machine learning applications but time-consuming and challenging.

Many excellent researches have done in this regard [1–4]. However, it seems there is a room of study in the perspective of domain, especially business domain. Data itself would have less meaning but it would have practical meaning when the data identified as important to the domain. When understanding the domain in business, identifying stakeholders and the stakeholders' goals is a critical factor in building reliable and meaningful software. Thus, understanding stakeholders and the stakeholders' goals can be a criteria in selecting data features. Our research started in this regard.

3 Goal-Oriented Data Feature Selection and Prioritization

In general, putting all data features collected as inputs of machine learning techniques does not guarantee good accuracy and meaningful insights, but rather harmful for getting the good accuracy and meaningful insights. Similarity, using fractional data features among the whole data features also has negative impacts on the good accuracy and insights. Selecting proper data features is essential process in machine learning for getting the better accuracy and meaningful insights, but difficult and time-consuming process.

$$\sum_{i=1}^{N} {}_{N}C_{i}$$

Fig. 1. The total number of combinations of data features

Figure 1 describes the total number of combinations of data features. N represents the total number of data features. For example, if N is 10, then the total number of possible combination is 1,023. Even just 10 data features can generate 1,023 different combinations of data features. To find a best machine learning model, we need to test many different machine learning algorithms and parameters on 1,023 different combinations. Practically, it is impossible because it is too time-consuming. If the number of data features are bigger, like 2,000 data features, the problems would be exaggerated. It seems a good way to select data features heuristically.

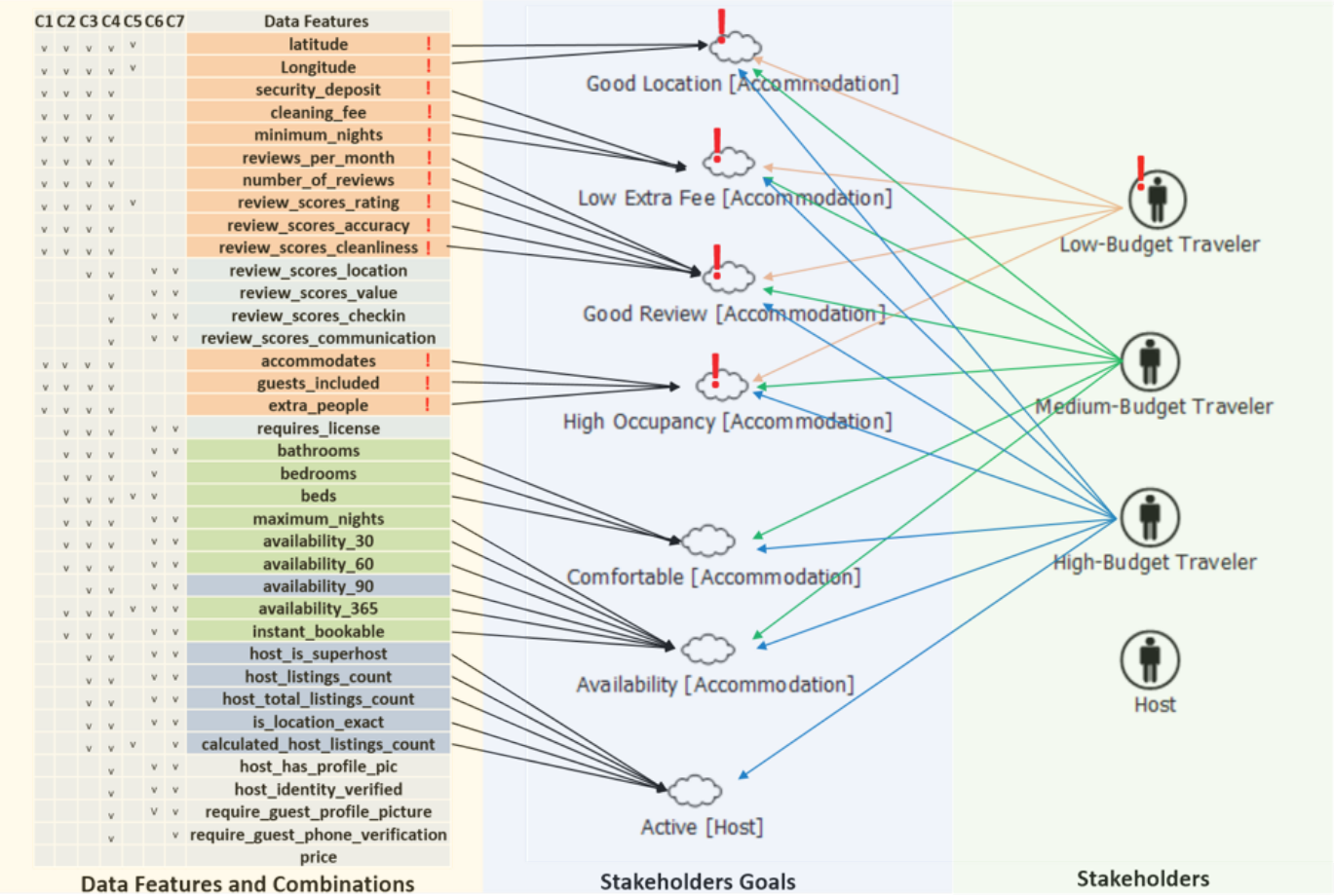

Fig. 2. Goal-Oriented Data Feature Selection Approach

Our approach is to select proper data features rationally and prioritize combinations of data features for testing systematically. To do that, we identify stakeholders and the stakeholders' goals that might have impacts on a task, and then identify relationships between the stakeholders' goals and data features. Data features that are related to important stakeholders' goals are considered as important data features. Combinations of data features are constructed for testing, and the combinations of data features that have more important data features are selected preferentially for testing. The purpose of our approach is not for identifying and prioritizing data features in a way of absolutely correct, but rather identifying and prioritizing data features rationally. This is due to the fact that it is impossible to know whether the data features selected are indeed important and whether a combination of data features can generate better accuracy and meaningful insights than different combinations of data features before testing.

However, an approach that can help in selecting data features rationally and in prioritizing combinations of data features can alleviate the difficulty and time-consuming problem in selecting data features.

Likewise, our example, identifying important stakeholders, the stakeholders' goals, and relationships between the important stakeholders' goals and data features, is not an absolutely complete example. Since real world is an open world, it is impossible to identify those properties completely and absolutely. There can be many different patterns of stakeholders, goals, and relationships. However, our example shows rational and systematic approach to identify those properties.

3.1 Stakeholders

Figure 3 shows stakeholders in Airbnb business. Stakeholder in Airbnb business is decomposed into two stakeholders; Traveler and Host. Host is a person who rent a room through Airbnb platform, and responsible for deciding price of the room. Travelers are further decomposed into three types of traveler; Low-budget traveler, Medium-budget traveler, and High-budget traveler.

Airbnb is an alternative supplier of lower-end hotels and hotels not catering to business travelers [8], thus we consider low-budget travelers as main target customers of Airbnb hosts.

3.2 Stakeholders and the Stakeholders' Goals

In coming up with selections of important data features using goal-oriented approach, identifying important stakeholders, their goals, and relationships between data features

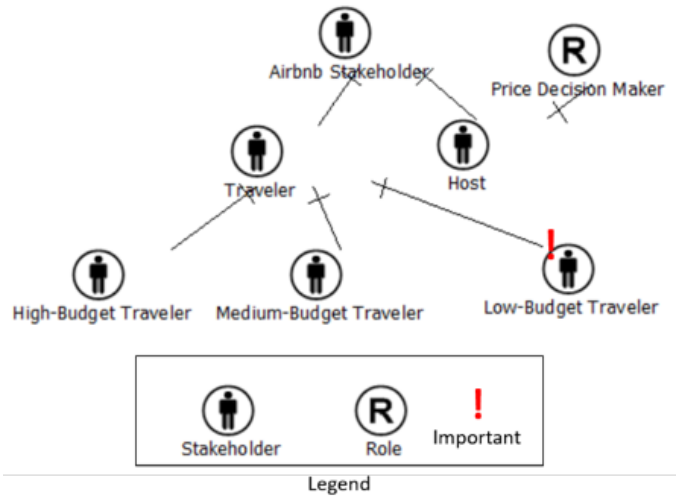

Fig. 3. Stakeholders in Airbnb Business.

and the goals is a foundation to select data features rationally. Figure 2 shows the big picture of the goal-oriented data feature selection approach.

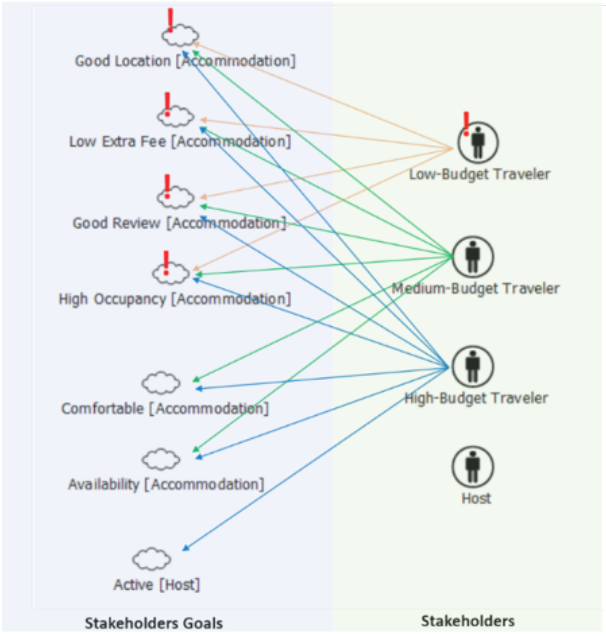

Fig. 4. Stakeholders and Stakeholders Goals

We identified low-budget traveler as an important stakeholder; consequently, low-budget customers' goals are considered as most important goals, and data features related to the most important stakeholder goals are considered as most important data features as well. Figure 4 describes stakeholders, their goals, and relationships between data features and the goals in our scenario.

In our scenario, low-budget travelers, marked as important, want to find an accommodation placed in a good location with low extra fee, good reviews, and high occupancy to share the room charge with as many people as possible. Also they are willing to abandon comfortableness of their travel, such as enough number of beds, bedrooms, and bathrooms, to lower their travel budget. Therefore, they have four goals, good location, low extra fee, good reviews, and high occupancy of an accommodation. The four goals are marked as important goals in Figure 4.

Medium-budget travelers are willing to plan more budget than low-budget travelers to find comfortable and popular room, thus they have comfortable and availability goals in addition to the four goals of low-budget travelers. In deciding comfortable rooms, the number of bedrooms, beds, and bathrooms are important features. Since the Airbnb hotel business is the domain of lower-end hotels, medium-budget travelers are considered as less important than low-budget travelers.

High-budget travelers afford to pay more budget than medium budget and low-budget travelers. In our scenario, they care the how active the host to estimate how much they can trust the room is. However, they might be more suitable for good hotels rather than lower-end hotel, so they are considered as less important than medium-budget travelers.

3.3 Stakeholders Goals and Features Selection

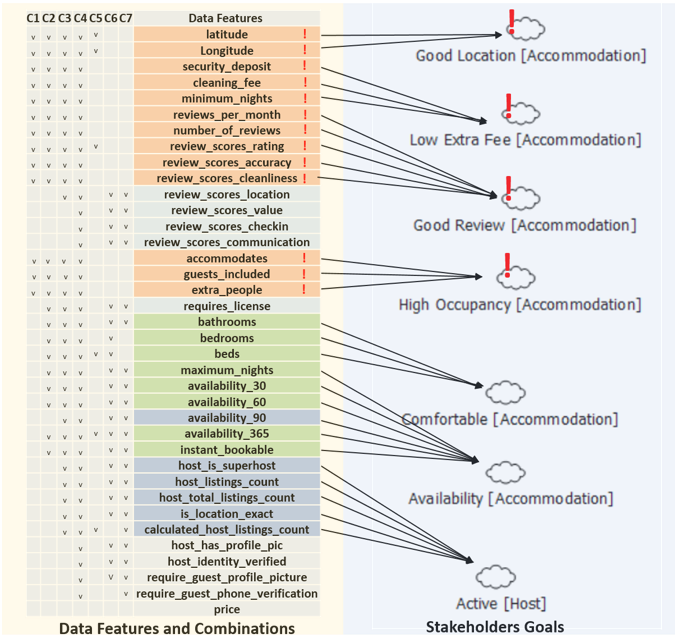

Fig. 5. Selecting Data Features according to Goals

Fig. 4 shows how to select data features according to goals. Latitude and longitude are related to good location goal, and good location is marked as important; consequently, latitude and longitude are considered as most important features. Likewise, important features related to important goals are marked with orange color and exclamation mark. Data features related to comfortable and available goals, second important goals, are marked with green color. Data features related to active goal, third important goal, are marked with blue color. The other features seem less related to goals or less critical to goals, thus the other colors are not marked with any color. In selecting data features, we select

important features, marked with orange color, with higher priority.

We construct multiple combinations of data features according to our goal-oriented feature selection approach, and the combinations are represented on left side of Fig. 4. C1 contains 13 data features only related to low-budget traveler goals. C2 contains 21 data features related to medium-budget traveler goals, and C3 contains 27 data features related to high-budget traveler goals.

C4 contains all the data features, C5 contains 7 fraction of all data features. C4 and C5 are constructed for comparing purpose with C1, C2, and C3. The comparison is to confirm whether data feature selection is indeed necessary by testing different number of data features.

C6 and C7 contains 21 data features that exclude data features marked as important. C6 and C7 are constructed to compare the performance in error metrics with C2, which contains same number of data features but contains data features marked as important.

We prioritize the data feature combinations according to the purity of the combinations. For example, C1 has only important features and C2 includes less important features in addition to important features. In this case C1 has higher priority, therefore C1 will be tested preferentially.

Since it is impossible to know which combination of features is better before testing the combinations, it is necessary to test all the combinations and see performance in error metrics. However, testing all the combinations is very time-consuming process thus we prioritize the data feature combinations to save time to test the combinations heuristically.

4 Experiment

4.1 Dataset

Airbnb of New York dataset [14] is used for price prediction problem in our experiments. 36 data features are extracted with 40587 instances from the dataset and price feature is used as a label for the features. Statistics of the price feature is described below.

Spark 2.1 with Scala 2.1 is used as a distributed computing platform for big data, and Spark MLlib is used as a machine learning library. Databricks [15] is utilized for computing platform for our experiments using Spark, Scala, and Spark MLlib. Databricks provides analytics platform on top of Apache Spark.

4.2 Experiment Modeling

Gradient-boosted tree regression is used for our price prediction problem. Iteration number as a parameter of the gradient-boosted tree regression model is changed from 10 to 50, and each iteration of the experiment runs a gradient-boosted tree regression model for 50 times. For example, in the first iteration, we run a model for 50 times with iteration parameter value 10. In the second iteration, we change the iteration number from 10 to 20, then run the experiment for 50 times.

The focus of our experiments is to show whether our approach, selecting and prioritizing data features according to important stakeholders' goals, can have positive impacts on

Label	Price
Number of Features	36
Number of Instances	40587

	Price
Average	145.12
Median	100
Standard Deviation	219.74
Max	10000
Minimum	10

Fig. 6. Dataset Statistics

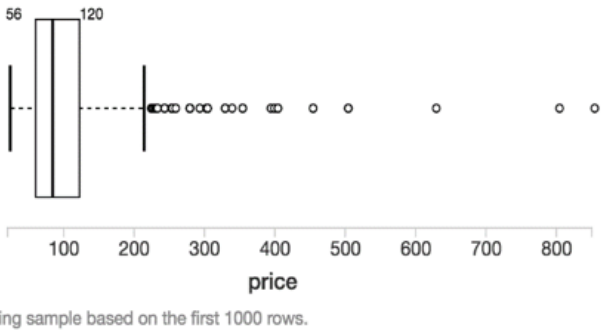

Fig. 7. Box Plot for Price

improving performance in error metrics. Thus, we run same experiments by just changing data feature combinations.

4.3 Experiment Result

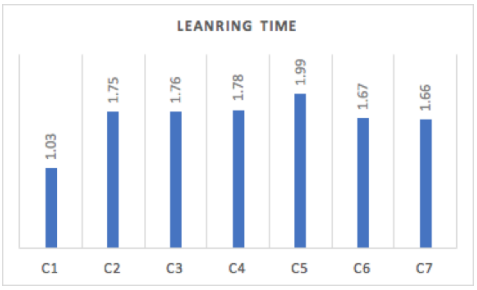

Fig. 8. Learning Time of Each Combination in Hour

Figure 8 shows learning time of each combination in hour and Figure 9 shows performance of each combination in error metrics. C4, containing all the features, is used as a baseline. C1, C2, and C3, containing data features selected by goal-oriented feature selection approach, show better

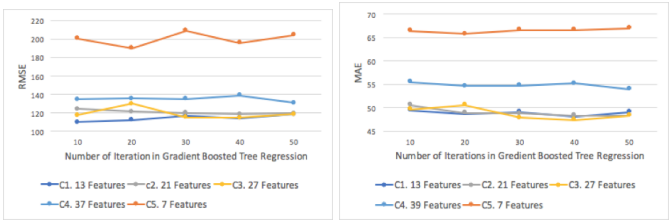

Fig. 9. Experiment Results of Each Combination of Data Features in RMSE and MAE.

performance in error metrics than C4 on each iteration. It shows that proper feature selection is helpful for getting better performance in error metrics. However, C5, containing too small number of data features randomly, shows worse performance in error metrics than C4.

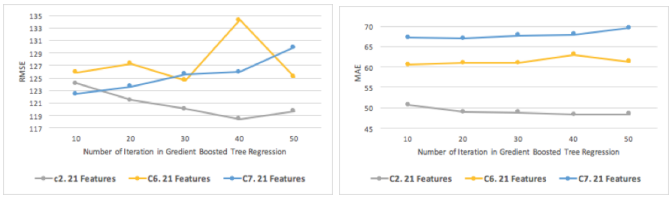

Fig. 10. Comparison of C2, C6, and C7 Containing 21 Data Features

To validate our approach in different perspective, we construct combination C6 and C7, having same number of data features with C2 but having different data features, then compare the performance in error metrics to C2. C6 and C7 exclude the data features, identified as important by our goal-oriented feature selection approach. Figure 11 describes that C2 shows lower RMSE and MAE than C6 and C7 as the number of iteration increases.

5 Observation and Discussion

5.1 Necessity of Feature Selection

Figure 9 also show the necessity of proper feature selection. C4, which is containing all data features, shows worse performance in MAE and RMSE than C1, C2, and C3, which are containing less number of data features. Using all data feature does not make better performance in MAE and RMSE in this experiments. However, C5, which is containing 7 fraction data features, shows worse performance than C1, C2, C3, and C4 in MAE and RMSE. It shows that too small number of data features make worse performance. Those experiments implies that proper and meaningful data feature selection is necessary to make better performance in error metrics. However, a question is how to select proper and meaningful data features. Our goal-oriented data feature selection approach can be useful to approach the problem and Figure 9 proofs it.

5.2 Performance Improvements

Figure 11 depicts performance improvements of the goal-oriented feature selection and prioritization approach by comparing C1, C2, C3, and C4 on each iteration. Compared to C4, C1 shows performance improvement from 9.44% to

Fig. 11. Performance Improvements C1, C2, and C3 Compared to C4

18.46% in RMSE and from 9.10% to 12.80% in MAE, C2 shows performance improvement from 7.91% to 14.68% in RMSE and from 8.78% to 12.51% in MAE, C3 shows from 9.5% to 17.22% in RMSE and from 7.46% to 14.10% in MAE.

	C1	C2	C3	C4
RMSE	109.9	118.4	114.88	130.93
MAE	48.15	48.31	47.43	53.96

Fig. 12. RMSE and MAE of C1, C2, C3, and C4

Figure 12 show best RMSE and MAE of C1, C2, C3, and C4. Combinations of data features selected by our goal-oriented data feature selection approach show better RMSE and MAE. Our experiments prove that our approach can yield better RMSE and MAE as the number of iteration increases, compared to the approach of using all data features.

5.3 Threat of Validity

We applied our approach in the domain of business, not an image processing or speech recognition. Data characteristics in business can differ from image or speech, thus it is hard to say our approach can be applicable to other domains. We need more research to figure out whether we can extend our approach to the other domains.

In our experiments, we used Gradient-boosted tree regression for price prediction by changing an iteration parameter. It seems necessary to run more experiments using more algorithms and parameters.

We used a heuristic approach to decide relationship between data features and goals, importance of data features, a purity of data feature combinations, and priority of combinations. It seems important to research more systematical approach.

6 Conclusion

The goal-oriented data feature selection approach proposed in this paper starts with the capturing and understanding of multiple stakeholders' goals, based on lower-end hotel business domain characteristics. Importance of each data feature is then identified based on those goals, as means of assessing the impact of feature selection on the degree to which the goals are satisfied. Different combinations of data features

are built based on those importance of each data feature. Priority is given to the each combination according to the purity of the combinations, degree of containing important and less important data features. Combinations containing more important data features without less important data features have higher priority. The prioritization helps in alleviating the time-consuming problem of testing all the combinations of data features. Combinations, containing important data features selected by our goal-oriented data feature selection approach, yield better performance in MAE and RMSE than the approach of selecting all data features or selecting data feature without important features.

Our approach demonstrates one way of exploring, identifying, evaluating, and selecting among data features with respect to stakeholder' goals. This is likely to provide better rational decision support to select data features and even better time saving for a huge number of testing combinations of data features, which seems to be among the most critical innovations for using machine learning techniques in business domain.

7 Future Work

We used Airbnb data to apply our approach to business domain. It seems necessary to apply our approach to more business domains. Our approach needs to be refined more in decomposing stakeholders' goals more, in deciding relationships between data features and goals more systematically, in quantifying the purity of data feature combinations, and in quantifying priority of the combinations. Technically, it seems necessary to experiment approach with more machine learning algorithms and parameters.

References

[1] Guyon, Isabelle, and Andr Elisseeff. "An introduction to variable and feature selection." Journal of machine learning research 3.Mar (2003): 1157-1182.

[2] Dash, Manoranjan, and Huan Liu. "Feature selection for classification." Intelligent data analysis 1.1-4 (1997): 131-156. Blum, Avrim L., and Pat Langley. "Selection of relevant features and examples in machine learning." Artificial intelligence 97.1 (1997): 245-271.

[3] Hall, Mark A. Correlation-based feature selection for machine learning. Diss. The University of Waikato, 1999.

[4] Mircea, M., Ghilic-Micu, B., & Stoica, M. (2011). Combining business intelligence with cloud computing to delivery agility in actual economy. Journal of Economic Computation and Economic Cybernetics Studies, 45(1), 39-54.

[5] Chung, Lawrence, et al. Non-functional requirements in software engineering. Vol. 5. Springer Science & Business Media, 2012.

[6] Chung, Lawrence, and Julio do Prado Leite. "On non-functional requirements in software engineering." Conceptual modeling: Foundations and applications (2009): 363-379.

[7] Supakkul, Sam, et al. "An NFR pattern approach to dealing with NFRs." Requirements Engineering Conference (RE), 2010 18th IEEE International. IEEE, 2010.

[8] Zervas, Georgios, Davide Proserpio, and John W. Byers. "The rise of the sharing economy: Estimating the impact of Airbnb on the hotel industry." Journal of Marketing Research (2014).

[9] Classification and regression, https://spark.apache.org/docs/2.1.0/ml-classification-regression.html

[10] Ensembles - RDD-based API, https://spark.apache.org/docs/2.1.0/mllib-ensembles.html

[11] White, Tom. Hadoop: The definitive guide. " O'Reilly Media, Inc.", 2012.

[12] Carbone, Paris, et al. "Apache flink: Stream and batch processing in a single engine." Data Engineering 38.4 (2015).

[13] Abadi, Martn, et al. "Tensorflow: Large-scale machine learning on heterogeneous distributed systems." arXiv preprint arXiv:1603.04467 (2016).

[14] Inside Airbnb- Adding data to the debate, http://insideairbnb.com

[15] Databricks, https://databricks.com/