Building Payment Classification Models from Rules and Crowdsourced Labels: A Case Study

Artem Mateush, Rajesh Sharma, Marlon Dumas(✉), Veronika Plotnikova, Ivan Slobozhan, and Jaan Übi

University of Tartu, Tartu, Estonia
{artem.mateush,rajesh.sharma,marlon.dumas,veronika.plotnikova, ivan.slobozhan,jaan.ubi}@ut.ee

Abstract. The ability to classify customer-to-business payments enables retail financial institutions to better understand their customers' expenditure patterns and to customize their offerings accordingly. However, payment classification is a difficult problem because of the large and evolving set of businesses and the fact that each business may offer multiple types of products, e.g. a business may sell both food and electronics. Two major approaches to payment classification are rule-based classification and machine learning-based classification on transactions labeled by the customers themselves (a form of crowdsourcing). The rules-based approach is not scalable as it requires rules to be maintained for every business and type of transaction. The crowdsourcing approach leads to inconsistencies and is difficult to bootstrap since it requires a large number of customers to manually label their transactions for an extended period of time. This paper presents a case study at a financial institution in which a hybrid approach is employed. A set of rules is used to bootstrap a financial planner that allowed customers to view their transactions classified with respect to 66 categories, and to add labels to unclassified transactions or to re-label transactions. The crowdsourced labels, together with the initial rule set, are then used to train a machine learning model. We evaluated our model on real anonymised dataset, provided by the financial institution which consists of wire transfers and card payments. In particular, for the wire transfer dataset, the hybrid approach increased the coverage of the rule-based system from 76.4% to 87.4% while replicating the crowdsourced labels with a mean AUC of 0.92, despite inconsistencies between crowdsourced labels.

1 Introduction

Understanding the expenditure patterns of private customers at a fine level of detail allows financial institutions to customize their offerings in order to address the diverse requirements of their customer base. A basic ingredient to build a deep understanding of expenditure patterns is to be able to classify Consumer-to-Business (C2B) payments across product categories (e.g. utilities, food, clothing,

R. Matulevičius and R. Dijkman (Eds.): CAiSE 2018 Workshops, LNBIP 316, pp. 85–97, 2018.
https://doi.org/10.1007/978-3-319-92898-2_7

electronics). However, C2B payment classification is a difficult problem because of the large and evolving set of businesses and the fact that each business may offer multiple types of products, e.g. a business may sell both food and clothing.

As in any other automated classification problem, there are broadly two approaches available: rule-based and machine learning-based. In rule-based payment classification, a set of rules is maintained (typically bootstrapped by domain experts) in order to map each payment record to a category. For example, a rule might state that all payments made to a given account (belonging to a telco) should be classified as "Utilities & Telecommunications". This rules-based approach is simple, but it requires rules to be maintained for every possible business, especially when the data is continuously gets updated with newer cases [15].

The alternative approach is to construct a machine learning model from a set of labeled payments. In order to have enough samples, a typical approach is to crowdsource the acquisition of the labeled data from the customers themselves. This crowdsourcing approach is hard to bootstrap as it requires a large number of customers to manually label their transactions for an extended period of time. Furthermore, indistinguishably similar transactions by different customers may have different labels, a phenomenon known as the noisy data problem [10].

In this study, we partnered with a financial institution which has an existing rule-based system in place for classifying transactions. A set of rules is defined to bootstrap a financial planning tools that allows customers to view their transactions. Specifically, transactions are classified using a two-level hierarchy of categories. At the bottom level, there are labels such as *grocery*, *restaurants & cafeteria* and *footwear*, for example. These bottom-level labels are called categories (66 categories in total) and are grouped into 14 *category groups*, such as *food*, *utilities & telecommunication*, *clothing*. Naturally, the defined rules are not able to classify every transaction. Accordingly, users of the financial planner are able to assign labels to the transactions that are left unclassified by the rule-based system. Additionally, users are able to re-label already classified transactions if they perceive that the assigned category is not correct.

After a few years of operations of the rule-based financial planning tool, the question arose of how to exploit the labeled data collected via this tool in order to build a more complete and accurate payment classification system to replace the existing rule-based one. This paper reports on the ensuing effort to construct an improved payment classification system, which combines the existing rule-based system with the crowdsourced labels collected via the financial planning tool.

Specifically, the paper describes the development of a payment classification model that integrates the following three sources:

1. **User-independent rules:** rules that map transactions to labels based on the beneficiary and (for card payments) the Merchant Category Classification (MCC) Code (cf. Sect. 3.2 for further details).
2. **User-defined rules:** These are rules defined by users, which assign labels to a transaction based on the payment's comment text or the beneficiary name. For example, a customer may define a rule that assigns a label *food* to every

transaction where the keyword "supermarket" appears in the transaction's comment.

3. **Manual user labels:** These are labels that are manually assigned by a user to a transaction. Manual labeling typically happens when the user disagrees with the rule-based labelling or when the rule-based labelling is not able to categorise the transaction in question. For example, if a customer visits a food shop to buy cooking utensils, the rule-based system will automatically assign the label *food* to this transaction. The user might then manually re-label this transaction to *household accessories*. In this case, the user does a one-off manual re-labelling rather than defining a general user rule.

To integrate the above three sources of labels, we trained a multiclass machine learning classifier from a dataset that combines samples labeled manually, samples labeled by user rules, samples labeled by user-independent rules, and samples that could not be labeled by any rule. As the resulting system combines knowledge originating from the crowdsourced labels as well as knowledge from the user-defined and user-independent rules, we call it a *hybrid classifier*. The paper presents an empirical evaluation of this hybrid classification approach in terms of coverage and accuracy over wire transfers and credit card transactions.

The rest of the paper is organized as follows. Section 2 presents related work. In the Sect. 3 we discuss the dataset used in this study, while in Sect. 4 we present the model training approach. In Sect. 5, we evaluate our approach, and in Sect. 6 we draw conclusions and outline future directions.

2 Related Work

In this section, we describe works from three different perspectives, namely (i) classification in crowdsourced data, (ii) classification in the noisy labels and (iii) payment classifications, at the intersection of which this work lies.

Classification from Crowdsourced Data. Various classification algorithms have been proposed for crowdsourced data [2] in applications such as twitter data for the traffic congestions [7], eateries [13] and medical data [4,12]. In particular, in the medical domain crowdsourced approaches have been used for validating machine-learning classifications [4,12]. Readers can refer to [8] for a comparative study of classification algorithms for crowdsourced data.

Noisy Labels. Noisy label problem has recently attracted a lot of attention from researchers [9,10,16]. In a theoretical study performed using synthetic dataset [10], authors presented a probability based solution to overcome noisy data problem. In another work [9], a framework based on distillation techniques has been presented. To handle the missing labels, a mixed graph framework is presented for multi-label classification in [16]. Most of these techniques have been applied and tested using image based datasets.

Payment Classification. Recently, there has been research related to the comparison of various classification algorithms such as SVM, neural networks, logistic regression for automatically classifying banking transactions [1,5,6,14]. Whereas the amount of the data being used for evaluation is not mentioned in [1,14] however, in comparison, the dataset used in the present study is much larger than [6]. In addition, these datasets did not suffer from the noisy data problem unlike that of ours.

Other related work includes existing approaches to use rule-based approaches for classification in Big Data settings, such as [15], which reports on the development of a system for classifying product items into product types at Wallmart-Labs. The authors note that in real-world classification problems, it is necessary to combine rule-based classification (handcrafted rules) with machine learning so as to maintain high precision (and improve recall) as the system evolves over time. The case study we report in this paper follows a similar approach, with the additional complexity that it relies on labels crowdsourced from customers, whereas the system in [15] relies on labels coming from crowdsourcing marketplaces, where workers can be prescribed with specific instructions on how to perform their manual classification task.

3 Datasets

This section describes the datasets of payments, payment classification rules, and manually assigned labels used for automated payment classification.

3.1 Payments Datasets

This dataset contains anonymized customers' transactions collected by the financial institution over the period of 10.5 months. The transactions are from three different Northern-European countries. For anonymity, we call the three countries as C1, C2, and C3. The dataset has two types of transactions. The first type which we call *account payments*, consists of transactions made via wire transfer, that is, transactions from one bank account to another. The second type, which we term as *card payments*, contains transactions between a bank customer and a business entity through the use of payment cards. Both of these payment types transactions can further be categorised into two dimensions. The first dimension consists of incoming and outgoing payments. The second dimension describes the type of counterparty, that is the party dealing with the customer of the financial institution. Table 1 provides the exact number of transactions in each of the cases in our dataset.

Account Payments. The *account payments* (AP) dataset describes transactions made between accounts, that is, wire transfers. It can be differentiated based on the type of the counterparty. The AP includes (i) person-to-person transactions within the financial institution (AP-P2P), (ii) person-to-business transactions within the financial institution (AP-P2B), (iii) person-to-financial

institution transactions (AP-P2F), and (iv) transactions outside the financial institution (AP-P2O), for which the financial institution does not have the information about one of the transacting parties. Table 1, columns P2P, P2B, P2F, P2O provide information about the number of transactions in each of the above cases. The nomenclature is based on the state of outgoing payments, but incoming payments are also present in each category. In P2P they are duplicates of the corresponding outgoing payments, and in P2B and P2F they represent cases such as salary payments, refunds and other forms of income.

Table 1. Dataset description (in millions)

Dataset	Country	Total Transactions			P2P			P2B			P2F			P2O		
Direction		I	O	Tot.	I	O	Tot.	I	O	Tot.	I	O	Tot.	I	O	Tot.
AP	C1	27.2	92.4	119.6	8.4	8.3	16.7	11.4	37.0	48.4	1.6	9.9	11.5	5.8	37.2	43.0
	C2	27.8	83.1	110.9	8.4	8.3	16.7	7.1	30.3	37.4	2.6	9.3	11.9	9.7	35.2	44.9
	C3	29.9	95.6	125.5	5.4	5.3	10.7	17.0	31.9	48.9	1.6	12.6	14.2	5.9	45.8	51.7
	Total	84.9	271.1	356.0	22.2	21.9	44.1	35.5	99.2	134.7	5.8	31.7	37.5	21.4	118.2	139.6
Dataset	Country	Total			-			CPPA			-			CPNA		
CP	C1	0.2	167.9	168.1	-			0.001	97.0	97.0	-			0.2	70.9	71.1
	C2	0.3	124.3	124.6				0.0005	35.1	35.1				0.3	89.2	89.5
	C3	0.4	116.0	116.3				0.1	46.9	47.0				0.3	69.1	69.4
	Total	0.9	408.2	409.1				0.1	179.0	179.1				0.8	229.2	230.0

Card Payments. Card payments (CP) represent the transactions made through a payment card (debit or credit). Based on the merchant that processes a transaction, we differentiate the *card payments* (CP) dataset into (i) card payments to merchants who have signed cards processing agreement (CPPA) with the financial institution with which we partnered for this study and (ii) card payments to the merchants that do not have the agreement with the financial institution of our study (CPNA)[1]. The internal information structure of the financial institution has a greater level of sophistication when it comes to transactions related to CPPA, which is the basis for our differentiation - as an example we are using CPPA transactions for augmenting our understanding about the focal businesses, when analyzing account payments. Like AP, the CP dataset also contains both incoming and outgoing payments.

3.2 User-Independent Rules

The pre-existing rule-based approach used in the financial planning tool of the financial institution is based on an ordered set of so-called user-independent rules. Each rule assigns a given label to a transaction if it fulfills certain conditions defined on the transaction's fields (e.g. account number of beneficiary, payment comment, etc.). Table 2 lists the user-independent rule types for each payment dataset in the order of priority, and the acronyms used to refer to them.

[1] AP-P2O contains transactions both to people and businesses outside but CPNA contains transactions only to businesses.

Table 2. Types of user-independent rules

Type	Dataset	Column
A	AP	Account number
C	AP, CP	Payment comment
R	AP, CP	Payment comment (regex)
I	AP	Internal type code associated
M	CP	Merchant Category Classification (MCC) code mapping

It is worth mentioning that the rule-based approach has a higher accuracy measure of the CP dataset compared to the AP dataset because of the external Merchant Category Classification (MCC) code field associated with the transactions. These codes categorize the payment under a different categorical hierarchy[2]. However, the rule-based mapping from the MCC to the two-level categorical hierarchy used in the financial institution leads to inherent mapping problems, as two different points of view on the consumption are considered. Additionally, MCC-based mappings introduce the problem of products heterogeneity, as a single card payment processing agreement only covers one MCC code, whereas multiple types of products are sold thereby.

3.3 User-Defined Rules and Manually Labeled Transactions

The reported payment classification case study had a scope limited to classifying consumer-to-business transactions, since these are most relevant when determining expenditure patterns. Given this scope, the dataset of transactions we took as input excluded the following categories:

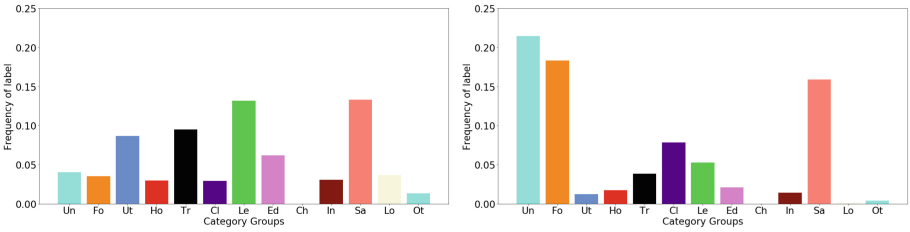
1. All incoming transactions (categorised as *income*)
2. P2P transactions (categorised as *private person payments*)
3. P2F transactions (categorisation already exists inside the financial institution in another context)
4. P2O transactions in AP (we have no way to separate person-to-person transactions)

In other words, the study reported here is limited to classifying outgoing P2B transactions in AP and outgoing CPPA and CPNA transactions in CP.

Originally, in our dataset there were 266 K manually labeled transaction records in AP and 71 K in CP. However, after limiting our scope to exclude the transaction categories mentioned above, the filtered dataset has only 50 K manual labels in the AP dataset and 40 K for the CP dataset. Figure 1 provides

[2] The description of the MCC hierarchy is available at https://usa.visa.com/dam/VCOM/download/merchants/visa-merchant-data-standards-manual.pdf.

the initial label distribution for the AP (Fig. 1(a)) and CP (Fig. 1(b)) datasets (refer to Table 5 for the acronyms being used in the Fig. 1.). It can be inferred that the customers tend to use wire transactions for *savings* and leisure & travel payment categories, while payment cards are most often used for the *food*.



(a) Accounts (b) Cards

Fig. 1. Label distribution of crowdsourced data

In addition to the manually labeled payments, we took as input 510 K user-defined rules created by 50 K customers. These rules work over the beneficiary field (account number of beneficiary) and/or over the payment comment. A typical user-defined rule might state example that payments made to a given company, and containing the keyword "catering" should be classified as *food*.

We applied these 510 K user-defined rules to the AP and CP dataset as an additional source of labels (to complement the user-independent rules and the manually assigned labels). We note that the user-defined rules have higher priority than the user-independent rules, meaning that if a given payment transaction matched both a user-defined rule (defined by the user who performed that transactions) as well as a user-independent rule, then the label given by the user-defined rule takes precedence (overrides) the label given by the user-independent rule. Similarly, the manually assigned labels have higher priority than the user-defined rules.

4 Model Training

We approach the problem of payment classification as a multiclass classification task, where each transaction has to be labeled with one out of 66 possible labels. For both AP and CP we exploit the following features: (1) identifier of the party, (2) transaction sum amount (log-transformed and normalized), (3) country, (4) id of counterparty bank, (5) vectorized payment comment text, (6) vectorized counterparty name text. Additionally, for AP dataset we use two more features (1) internal codes determining some transaction types and (2) labels transferred from card payments for this party, and in the CP dataset an additional feature of code is used.

If we rely only on the manually assigned labels as true labels for the purpose of training the classifier then, two problems arise which make these labels insufficient. Firstly, the number of manually assigned labels (50 K) is too small (by three orders of magnitude) compared to the size of the total dataset. Secondly, the set of manually assigned labels is non-representative by definition (not all transactions are labeled, only ones where customers are unsatisfied with automatic labeling). To prevent these issues from affecting our model's performance, we **enrich** the dataset used for training the model by adding transactions where we are confident that the labels being assigned by the rules are correct. In order to select these transactions, we limit ourselves to the transactions that belong to the customers who use the online system. The reason behind this decision is based on the fact that the rule-based labels are seen by those customers who have opted not to change the labels, which guarantees their correctness. We augment the dataset with 3 additional samples of size equal to the size of the original dataset, which consist of:

1. transactions with labels produced by user-independent rules.
2. transactions with labels produced by user-defined rules;
3. transactions without labels.

We trained the classifier over a dataset consisting of equal shares of labels from each of these sources. Since the smallest source is the first one (manually assigned labels), we took all transactions from this source (50 K for AP, 40 K for CP) and we randomly extracted equally sized samples from the other three sources.

We used the XGBoost classifier for training the model. The selection of this particular classifier has been motivated by its performance in previous evaluations such as [11], where it outperformed other classifiers such as random forest [3]. Having trained a classifier from the above combination of labeled samples, we combine it with the pre-existing rule-based classification system as follows. If the XGBoost classifier manages to assign a non-null label to a given transaction (in the testing set), we keep this label. If it assigns a null label to a transaction, but there exists a user-independent rule that assigns a non-null label, we use the label assigned by the user-independent rule. If neither the XGBoost classifier nor the user-defined labels can classify the transaction, we leave it with a null label. We call the resulting combined classifier the *hybrid classifier*.

5 Evaluation Results

This section describes the results of our proposed classifier compared to the existing rule-based system. We access the quality of our classier in terms of three performance measures, namely (1) coverage, (2) AUC score and, (3) overriding score. In the following three subsections, we first define these measures before discussing the evaluation results for each of them.

5.1 Coverage

We measure the coverage in order to understand to what extent our proposed model covers the set of whole transactions compared to the baseline model. We define coverage as the percentage of transactions to which a model (rule-based and hybrid) can assign a known label and is formally defined as $Cov = \frac{N_+}{N}$, where N_+ is the number of non-zero labels and N is the total size of the dataset. For the hybrid model, in case the ML component is not able to assign a label then, a rule-based label is taken into consideration.

Table 3. Coverage scores of the classifier per group

Dataset	Coverage for rule-based model	Coverage for hybrid model
AP	76.4%	87.4%
CP	99.2%	99.8%

We calculate coverage on a random sample of 200,000 transactions from the whole dataset. Table 3 provides the information about the coverage being observed in both AP and CP datasets for the existing approach being employed by the financial institution (Column 1) as well as for our proposed hybrid approach (Column 2). We can see the improvement in all cases by using our proposed classifier. Particularly, in the case of AP dataset, the model has achieved an improvement of 11%, which is a significant improvement.

5.2 AUC Score

In addition to coverage, we also measure accuracy to evaluate how well the labels can be predicted using our hybrid model compared to the true labels. Because of the presence of class imbalance (some classes are rare), we measure accuracy by the means of AUC score.

In line with standard practice, we used 5-fold cross-validation to calculate the AUC scores. In each iteration, we train the model on 80% of the samples in the dataset and then we validate our model on the remaining 20%. All the reported AUC values are averaged over 5 folds. Also, we applied hyper-parameter optimization using grid search. Specifically, we varied two key hyper-parameters of XGBoost: the learning rate from 0.05 to 0.2 in steps of 0.05 and the maximum tree depth from 4 to 16 in steps of four. We selected the combination of hyper-parameters that led to the highest AUC averaged over 5 folds.

AUC is normally defined in the context of binary classification, but in our case we are dealing with a multi-class classification problem. Accordingly, we calculate the AUC for each class separately, and then we aggregate the class-specific AUCs into a total AUC measure. Specifically, we define total AUC as $AUC = \sum_{i=1}^{L} p_i \frac{N^i}{N_+} AUC_b^i$, where L is the number of labels (65 without "unknown"), N^i

is the number of i_{th} labels in the training set, N^+ is the number of labels (without "unknown") in the training set. AUC_b denotes a binary AUC function for i_{th} label and is defined as $AUC^b = \int_{\infty}^{-\infty} TPR^i(T)\left(-FPR^{i'}(T)\right)\, dT = P(X_1^i > X_0^i)$, where TPR is true positive rate, FPR is false positive rate, X_1, X_0 are the events that correspond to i_{th} label having true and false labels.

Table 4 presents the total AUC scores for AP and CP datasets for two cases: (1) without enrichment, i.e., when we only use manually labeled transactions and (2) with enrichment, i.e., by also including a sample of the rule-based labels in our training set.

Table 4. AUC scores with and without enrichment

AUC score	AP	CP
Without enrichment	0.81	0.80
With enrichment	0.92	0.98

As expected, the AUC on the dataset without enrichment is lower than the AUC on the dataset with enrichment. This happens because in the dataset without enrichment there are less regularities – none of the labels comes from used-defined rules, but only from manually assigned labels, which in addition are sometimes inconsistent with others. In contrast, the enriched dataset has more regularity because part of the dataset is labeled using rules. Meanwhile, the lower score for CP in the non-enriched dataset is justified by the fact that the manual relabeling in CP occurs only when rule-based labels are wrong as compared to AP, when a manual label can also be assigned to transactions with no rule-based labels which are more populous in AP.

Table 5. AUC scores over the individual category groups

Code	Category group name	Without enrichment		With enrichment	
		AP	CP	AP	CP
Fo	Food	0.88	0.82	0.95	0.99
Ut	Utility, telecommunication	0.88	0.71	0.95	0.99
Ho	Household	0.77	0.75	0.90	1.00
Tr	Transportation	0.90	0.91	0.96	0.99
Cl	Clothing	0.83	0.85	0.93	0.99
Le	Leisure, travelling, spare time	0.66	0.78	0.85	0.97
Ed	Education, healthcare, beauty	0.78	0.85	0.91	1.00
Ch	Children	0.76	0.84	0.90	0.99
In	Insurance	0.98	0.76	0.99	0.96
Sa	Savings and investments	0.85	0.83	0.94	0.86
Lo	Loans and financial services	0.92	0.82	0.97	1.00
Ot	Other	0.67	0.73	0.83	0.97

Table 5 provides information about the averaged AUC score for each category. We observe that for categories like Insurance with a small number of merchants and regular payments the AUC over manually labeled dataset is high.

5.3 Overriding Score

We also measure the difference between hybrid model and rule-based label's output for the same set of transactions after the dataset enrichment. To do that we define an overriding measure, that showcases the changes in the hybrid model's prediction compared to the rule-based model's prediction for the same transaction. We define overriding measure as $Ov = \frac{N_{dif}}{N_+}$, where N_{dif} denotes the number of cases where the hybrid model predicts a different label compared to the rule-based model and, N^+ represents the number of known labels.

This overriding measure essentially captures the refinement that our proposed model has introduced over rule-based approach. We can consider this score as a measure of refinement due to the fact that the ML model's output learns exceptions to the rules from manually labeled dataset, and thus, enhancing the predictive capability of the hybrid system over rule-based approach. This is based on the fact that manual labels represents the ground truth as they have explicitly overridden the rule based output. The instances where the ML model outputs no label are filled assigned using the rule-based labels, thus, they count for N^+ and not in N_{dif}. It is computed on the enriched dataset used for training. We achieve an overriding score of 26.4% on the AP dataset and 11.9% on the CP dataset, which indicates a high level of improvement over the existing rules.

5.4 External Validation

To complement the validation reported above, we conducted a small-scale validation with the help of six employees of the financial institution. The employees classified their own personal transactions during a one-month period (subsequent to the period covered by the dataset used for training the model). The resulting dataset consists of 109 labeled payments.

To measure model accuracy on this dataset, we use the *hit ratio measure*, i.e. the percentage of transactions to which a model (rule-based and hybrid) assigns a correct most likely label. It is formally defined as $Acc = \frac{TP+TN}{N}$, where N is the total size of the dataset, TP is the number of true positive labels and TN is the number of true negative labels. The rule-based classifier achieves a hit ratio of 39%, while the hybrid classifier scores 56%, which shows a major improvement. We acknowledge that the small size of the dataset is a threat to validity. On the other hand, this external dataset is free from the potential biasing and reliability concerns related to the assignment of labels.

6 Conclusion

In this paper, we presented a hybrid approach, which exploits rule-based system as well as the crowdsourced data provided by customers, to automatically classify C2B payments. We evaluated our model on a real but anonymised dataset

consisting of customers' transactions across three Northern-European countries and consists of two transactions types: (1) wire transfers (AP) and (2) card payments (CP). On the AP dataset, our model achieves an AUC of 0.92 and achieves an improvement of 11% in coverage, and provides overriding of 26.4%, compared to the existing rule-based approach. On the CP dataset, our model achieves an AUC of 0.98 and achieves a slight improvement of 0.6% in coverage, as well as providing overriding of 11.9%, compared to the existing rule-based approach.

We have multiple future directions for this work. We would like to investigate the problem using larger dataset as well as including user created text based rules as an additional feature in our model, which will allow us to measure improvements more clearly. We also plan to perform external validation using a larger real labeled dataset in order to check the accuracy of our model.

Acknowledgments. This work is supported by an unnamed financial institution and the European Regional Development Funds. We thank the employees of the financial institution who volunteered to create the external validation dataset.

References

1. Bengtsson, H., Jansson, J.: Using classification algorithms for smart suggestions in accounting systems. Master thesis, Chalmers University of Technology Gothenburg, Sweden (2015)
2. Bonald, T., Combes, R.: A streaming algorithm for crowdsourced data classification. CoRR, abs/1602.07107 (2016)
3. Chen, T., Guestrin, C.: XGBoost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD, pp. 785–794 (2016)
4. Duda, M., Haber, N., Daniels, J., et al.: Crowdsourced validation of a machine-learning classification system for autism and ADHD. Transl. Psychiatry **7**(5), e1133 (2017)
5. Etaiwi, W., Biltawi, M., Naymat, G.: Evaluation of classification algorithms for banking customer's behavior under apache spark data processing system. Procedia Comput. Sci. **113**, 559–564 (2017)
6. Folkestad, O.E.E., Vollset, E.E.N.: Automatic classification of bank transactions. Master thesis, Norwegian University of Science and Technology, Trondheim (2017)
7. Kurniawan, D.A., Wibirama, S., Setiawan, N.A.: Real-time traffic classification with twitter data mining. In: 2016 8th International Conference on Information Technology and Electrical Engineering (ICITEE), pp. 1–5, October 2016
8. Lesiv, M., Moltchanova, E., Schepaschenko, D., et al.: Comparison of data fusion methods using crowdsourced data in creating a hybrid forest cover map. Remote Sens. **8**(3), 261 (2016)
9. Li, Y., Yang, J., Song, Y., et al.: Learning from noisy labels with distillation. CoRR, abs/1703.02391 (2017)
10. Natarajan, N., Dhillon, I.S., Ravikumar, P.K., Tewari, A.: Learning with noisy labels. In: Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 26, pp. 1196–1204. Curran Associates Inc. (2013)

11. Nielsen, D.: Tree boosting with XGBoost. Master's thesis, NTNU, Trondheim, Norway (2016)
12. Noren, D.P., Long, B.L., Norel, R., et al.: A crowdsourcing approach to developing and assessing prediction algorithms for AML prognosis. PLoS Comput. Biol. **12**(6), e1004890 (2016)
13. Salehian, H., Howell, P., Lee, C.: Matching restaurant menus to crowdsourced food data: a scalable machine learning approach. In: Proceedings of the 23rd ACM SIGKDD, pp. 2001–2009 (2017)
14. Skeppe, L.B.: Classify Swedish bank transactions with early and late fusion techniques. Master thesis, KTH, Sweden (2014)
15. Suganthan, P., Sun, C., Gayatri, K.K., et al.: Why big data industrial systems need rules and what we can do about it. In: Proceedings of ACM SIGMOD, pp. 265–276 (2015)
16. Wu, B., Lyu, S., Ghanem, B.: ML-MG: multi-label learning with missing labels using a mixed graph. In: IEEE ICCV, pp. 4157–4165, December 2015