Demand Forecasting for Industry 4.0: predicting discrete demand from multiple sources for B2B domain

Jože Martin Rožanec†
Qlector d.o.o.
Jožef Stefan Institute International
Postgraduate School
Ljubljana, Slovenia
joze.rozanec@qlector.com

Dunja Mladenić
Jožef Stefan Institute
Jožef Stefan Institute International
Postgraduate School
Ljubljana, Slovenia
dunja.mladenic@ijs.si

Blaž Fortuna
Qlector d.o.o.
Jožef Stefan Institute International
Postgraduate School
Ljubljana, Slovenia
blaz.fortuna@qlector.com

ABSTRACT

Demand is the amount of certain product required by buyers at a point in time. Demand forecasting tries to predict future demand based on available information. It is considered a key component of each manufacturing company since improvements on it translate directly to resources planning, stocks and overall operations.
In the context of Industry 4.0, industry digitalization provides an ever-increasing number of data sources which can be consumed to gain visibility over all operations and used to optimize different processes within it. This also opens new possibilities into the field of demand forecasting, where multiple data sources can be integrated to get timely data for accurate forecasts.
We describe an efficient approach for demand forecasting for discrete components B2B industry. The proposed approach provides as good or better forecasts as logisticians for most months in six months period and achieves savings considering all test months period.

CCS CONCEPTS

• Information systems → Information systems applications → Enterprise information systems → Enterprise resource planning • Computing methodologies → Machine learning →Machine learning approaches
• Computing methodologies → Artificial intelligence

KEYWORDS

demand forecasting, industry 4.0, B2B manufacturing, time series analysis

ACM Reference format:

Jože Martin Rožanec, Dunja Mladenić and Blaž Fortuna. 2019. Demand Forecasting for Industry 4.0: predicting discrete demand from multiple sources for B2B domain. In *Proceedings of SiKDD (SiKDD'19). IS, Ljubljana, Slovenia.*

1 INTRODUCTION

Demand forecasting is the task of predicting the number of units of a specific good for a given point of time in the future before we actually get all orders from the customers. It is a critical factor in just-in-time supply chains, where companies are expected to offer short lead times for products with complex production processes made of raw materials or components with longer lead times. In this paper we focus on solving this task by using machine learning techniques.
As a socioeconomic phenomenon there are many aspects that may enhance predictions when captured into features, such as economic context (does demand increase with economic growth, how it is affected by price changes, are there substitute products, what kind of market do we operate on), other context facts (marketing campaigns, fashionable features, product established in market or a new release) or inherent product properties (product category, whether is perishable, etc.). By considering a wider context, we may mitigate demand signal distortions that happen at each new intermediary level of a supply chain, in what is known as the bullwhip effect [1]. Another factor of uncertainty is the forecasting horizon: further the horizon, less likely is to be the future similar to past and present state of matters and more difficult to be predicted accurately [2].
When considering forecasting techniques, it may be important to consider characteristics of demand. Authors discriminate demand along two main variables: by considering variability in demand timing and quantity. A classification scheme is described in *Figure 1*.

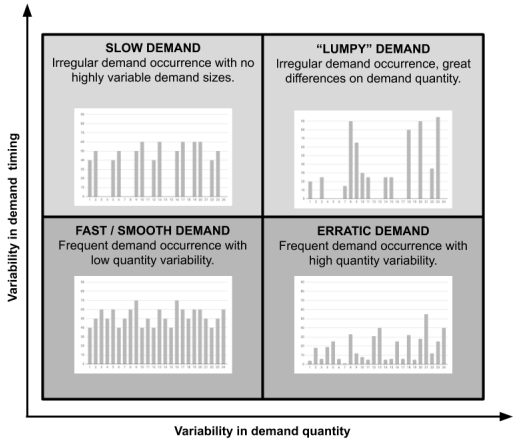

Figure 1: demand classification as per Williams et al. [3] and further elaborated in [4]

In our case we focus on demand forecasting for items from the B2B discrete manufacturing industry which are established in the market and sold under perfect market conditions. Since most of the products correspond to fast moving inventory, we do not discriminate between different demand types and treat all the products in the same way.
Publications addressing demand forecasting explored auto-regressive moving averages [5, 6, 7, 8], multiple linear regression [9] (MLR), Bayesian approaches [10], support vector regressors (SVR) [11] and artificial neural networks (ANN) [12]. In our research we consider naïve forecasting (last observed value as prediction), auto-regressive moving average (ARIMA), MLR, SVR and gradient boosted regression trees (GBRT) [13] models and compare them to logisticians predictions issued in two points in time: six weeks and three days before the event. We do not consider ANNs due to our limited amount of available data to train them.
The remainder of this paper is structured as follows. In Section 2, we define the problem, features, metrics and briefly describe forecasting techniques. Section 3 describe our dataset and preprocessing steps. In Section 4 we describe the experiments we conducted and results we obtained. Section 5 presents conclusions and directions for further work.

2 PROBLEM DEFINITION

Demand forecasting requires to predict the number of units for a product that will be ordered at a given future point in time. We consider different time horizons: H_1, … H_n, and train a specific model for each of them. The goal is to make accurate predictions about future demand based on historical demand data, annual sales plans, open sales orders and some contextual information.

2.1 Features

The main features we obtain from datasets correspond to historical data describing observed demand for each product, high-level estimations such as annual forecasts (describe expected product demand over the year), low-level demand proxies such as open sales orders for a given point in time, and contextual data (economic indicators, prices for relevant raw materials, vacation periods at buyers and manufacturer companies).

Derivative features are meant to explore the relation between the original variables as well as how do they relate to each other in different points in time. This way they reflect the direction and magnitude of trends in comparison to previous months. Months immediately before the target date provide information about recent demand and context behavior, while values from the same months but considered a year before help to learn seasonality patterns where it may exist.

The annual sales forecast and open sales orders give us some insight to the expected future. The annual forecast displays total amount to be sold over the year and a projected sales distribution. Open sales orders give us a weak signal about expected demand and may help to better estimate the target value given the rest of the feature's context. Both can also be related to learn if projected sales accurately reflect the annual forecast, differ by some factor or may not follow original expectations at all. In a similar way we learn past relations between projected and real demand as well as the relation between open sales at a given point in time and later demand realization.

Since we have two forecasting horizons with a six weeks separation and data available at a monthly frequency, we are able to compute additional features for models aimed to predict three days before the event horizon.

2.2 Metrics

To measure forecast performance across models we chose the mean absolute error metric. This metric is not sensitive to occasional large errors, which is important in the context of demand forecasting, where at specific points in time demand may display abnormal behavior that cannot be forecasted. The model should not be strongly penalized on them when trained. The metric also provides a straightforward interpretation (errors are measured in the same units as data and error magnitudes directly correlate on how well/bad the model performs). This does not turn into an issue when comparing different models, since by working on same dataset, we measure all models in same units and magnitudes.

We use the same metric as objective and evaluation metric for models we train.

2.3 Prediction techniques

We take into account five types of forecasting techniques: naïve forecasting, autoregressive integrated moving average (ARIMA), multiple linear regression (MLR), support vector regressor (SVR) and gradient boosted regression trees (GBRT). ARIMA and MLR are widely used in the literature to forecast fast moving products, while gradient boosted regression trees, to the extent of our knowledge, were not applied to demand forecasting in the B2B manufacturing industry.

Naïve forecasting method considers that the value to take place at time t+1 will be close to the one present at time t and thus the best proxy is to use the same value of time t as prediction. In our case we consider the last demand value we are able to observe given a time horizon as the output value of our prediction.

ARIMA is a stochastic time series method that grounds its predictions on three components: auto-regression (estimates white noise affecting the data by regressing the variable on own past values), integration (reduction of seasonality and trend by differencing the time series) and moving average (considers previous values to estimate the target value).

Both, the naïve forecasting and ARIMA are limited only to demand forecasting historic values and cannot consider a broader context in their predictions.

MLR is a simple method that explains linear relationships between a continuous dependent variable and multiple independent ones. The independent variables may be continuous or one-hot encoded categorical ones.

SVR is a regression method based on support vectors, where a kernel is used to map low dimensional data into a higher dimension and then best hyperplane and boundary lines are computed to predict target values. The method allows to fit the error within a certain threshold. In our case we use a radial basis function kernel (RBF kernel), which helps us to consider non-linear relationships between features.

GBRT makes use of gradient boosting, which generalizes boosting to an arbitrary loss function, and uses regression trees to approximate the negative gradient. These are built iteratively, each tree representing a step of gradient descent when optimizing the loss function.

3 DATA DESCRIPTION

3.1 Dataset

Our dataset was provided by manufacturing B2B industries and contains information about 69 products over a period of 68 months.

Among features we have historic demand data for all products, annual demand plans and open sales orders when the forecast is issued. Our prediction target is the amount of a certain product to be demanded by buyers for a given month - on two prediction horizons: six weeks and three days ahead.

3.2 Data preparation

Given the original dataset, we first analyzed data density. We found that there are multiple products with scarce demand datapoints due to irregular demand or by the fact that started being produced later in time. Since demand points density may affect model results, we decided to create multiple datasets based on how many points of historical demand data do we have - all with identical features. This way for all experiments performed, we have datasets with 0+, 10+, 20+, 30+, 40+ and 50+ demand history points.

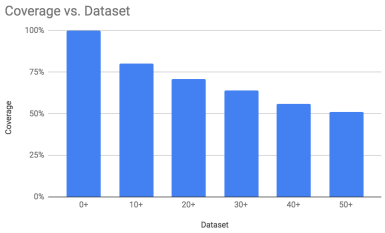

Figure 2: product coverage by dataset.

We then analyzed data distributions and observed that most features display Normal distributions when considering a single product, but over the whole dataset the distribution is lognormal. To mitigate this issue and differences in orders of magnitude, we first transformed them using a Yeo-Johnson transformation followed by standard scaling and a Min-Max transformation. The Yeo-Johnson transformation [14] ensures transformed values follow a Gaussian distribution, while the standard scaling centers them around zero with a standard deviation of one. By using the Min-Max transformation we get them into [0-1] range regardless of their original magnitudes. We also observed that some materials exhibit seasonality and trend but did not perform any ad-hoc preprocessing for them.

Among computed features, there are many that refer to past performance (same month or months close to it, for current year as well as the year

before). This cannot be computed where we lack enough history and thus decided to compute them and then prune the dataset to last 56 months to discard spurious values.
Considering that demand forecasting models are time sensitive, we use last six months for testing and devote the rest to train the models. We do so in such a way that the train set is not fixed, but we use all data up to the month to be predicted for training. By doing so, we had more records available to train models targeted towards last months and could ensure time proximity towards them.
We devote a month close to the test set as validation set. We performed an experiment to understand if excluding validation set the data from the train set affects results by degrading predictions or if including it causes the model to overfit. Results showed that including the validation set into the training set improved results without risk of overfitting and thus used this setup for the experiments.
The dataset with all features described is used for the MLR and GBRT models, while the naïve and ARIMA models use only historic demand data for a given product up to the month when the prediction shall be made.

4 EXPERIMENTS AND RESULTS

All experiments above were performed on datasets with demand records density of 0+, 10+, 20+, 30+, 40+ and 50+ records, to understand the tradeoff between data completeness and a greater number of records reflects in forecast results. In all cases we devote last six months to testing, and the rest of the data to train the model.
We use the following notation to describe models: *ModelName-FeatureSet-Transform-DatasetFiltering*
Valid *ModelName* values are SVR, MLR and GBRT; *FeatureSet*s can be 3m, 6m, 9m and 12m – notating that features were computed over a window of three, six, nine or twelve months. *Transform* can be "*wTT*" if transforms were applied to dataset target, otherwise we use "*nTT*". *DatasetFiltering* accepts three possible values: "*2Y*", "*3Y*" or "*4Y*" indicating that the dataset contains train records for two, three or four years respectively plus six months of test data.
Results are expressed in error ratio, computed as:

$$Model\ Error\ Ratio = 1 - \frac{MAE\ Model}{MAE\ logistician}$$

4.1 Feature set comparison

First experiment we performed was to understand how many months we should consider when looking back to create features in order to make better predictions. To this purpose, we developed four sets of features, created with a time window of three, six, nine and twelve months from target date.
When considering a six weeks horizon, we found out that best results were achieved by GBRT-9m-nTT-4Y and GBRT-6m-nTT-4Y, followed by GBRT-3m-nTT-4Y which accounts for half of second-best predictions. For a three-day horizon, most best results were achieved by GBRT-3m-nTT-4Y, making best prediction for half of datasets and second-best prediction for two of three remaining ones.

FIRST BEST				
Dataset	Feature set	Records time span	Target transform	Model error ratio
0+	9m	4Y	NO	0.11
10+	6m	4Y	NO	0.15
20+	12m	4Y	NO	0.11
30+	9m	4Y	NO	0.14
40+	6m	4Y	NO	0.14
50+	6m	4Y	NO	0.17

Table 1: best results when considering six weeks forecasting horizon. All of them run with GBRT algorithm.

FIRST BEST				
Dataset	Feature set	Records time span	Target transform	Model error ratio
0+	3m	4Y	NO	0.12
10+	3m	4Y	NO	0.17
20+	6m	4Y	NO	0.09
30+	3m	4Y	NO	0.15
40+	12m	4Y	NO	0.19
50+	9m	4Y	NO	0.17

Table 2: best results when considering three days forecasting horizon. All of them run with GBRT algorithm.

4.2 Target normalization

We then compared trained GBRT models against new ones where same transformations as applied to features were applied to target values. Our assumption was that by transforming the target, which had a lognormal distribution, we should get a better spread of predictions and better results.
Most best results for six-weeks horizon were found at GBRT-6m-wTT-4Y and GBRT-9m-wTT-4Y models except for 10+ and 50+ datasets. When comparing models with and without target transform, most best results at models without target transform resulted in second best results if considered globally.
On the other hand, for three-day forecasting horizons, applying transformations to the target improved results most cases, but still half of best predictions could be found among models that do not require target transformation. In this context, GBRT-12m-wTT-4Y displayed best global performance for half of datasets considered.

FIRST BEST				
Dataset	Feature set	Records time span	Target transform	Model error ratio
0+	9m	4Y	YES	0.16
10+	12m	4Y	YES	0.18
20+	6m	4Y	YES	0.12
30+	6m	4Y	YES	0.15
40+	9m	4Y	YES	0.15
50+	9m	4Y	NO	0.17

Table 3: best results when considering six weeks forecasting horizon. All of them run with GBRT algorithm.

FIRST BEST				
Dataset	Feature set	Records time span	Target transform	Model error ratio
0+	12m	4Y	YES	0.15
10+	3m	4Y	NO	0.17
20+	12m	4Y	YES	0.11
30+	12m	4Y	YES	0.19
40+	12m	4Y	NO	0.19
50+	9m	4Y	NO	0.17

Table 4: best results when considering three days forecasting horizon. All of them run with GBRT algorithm.

4.3 Records history contribution

Since forecasting models are time sensitive, we explored if recent history is more relevant in such a way that older records may deteriorate forecasting results. We pruned the dataset removing all records older than two or three years in train set and compared models trained on them with those obtained from training on full history.
When analyzing a six-weeks horizon, we found out that pruning history leads to better results achieving almost all first- and second-best results globally. Best results were achieved by models with three years of history with best performance for GBRT-9m-wTT-3Y.
For a three-days horizon, we observed that GBRT models with different feature sets over pruned datasets performed worse than existing ones. Overall, we observe GBRT algorithm achieved best results with target transforms enhancing results on half datasets and that 12m was the most frequent feature set among competitive models.

FIRST BEST				
Dataset	Feature set	Records time span	Target transform	Model error ratio
0+	9m	4Y	YES	0.16
10+	12m	3Y	YES	0.22
20+	12m	2Y	YES	0.17
30+	9m	3Y	YES	0.18
40+	9m	3Y	YES	0.20
50+	3m	2Y	YES	0.20

Table 5: best results when considering six-weeks forecasting horizon. All of them run with GBRT algorithm.

FIRST BEST				
Dataset	Feature set	Records time span	Target transform	Model error ratio
0+	12m	4Y	YES	0.15
10+	3m	4Y	NO	0.17
20+	12m	4Y	YES	0.11
30+	12m	4Y	YES	0.19
40+	12m	4Y	NO	0.19
50+	9m	4Y	NO	0.17

Table 6: best results when considering three-days forecasting horizon. All of them run with GBRT algorithm.

4.4 Comparison against models from literature

In literature most cited models were naïve, ARIMA, MLR and SVR, being SVR the one with state of art results. We trained MLR and SVR under same conditions as our best models, to understand how compare against them.
For a six-weeks horizon, we observed that GBRT outperformed them in all cases. Results are consistent with descriptions from literature, where ARIMA estimates are better than the naïve forecast, but surpassed by the SVR model in all cases. SVR and MLR consistently displayed best results with features computed over last three months regardless of dataset pruning, but MLR shows a rapid prediction quality degradation on the rest of feature sets. Despite this, best results were delivered by MLR over SVR.
Results for three-days horizon were similar. MLR had worst results when using 9m or 12m feature sets, followed by naïve forecasting. MLR and SVR displayed best results for 3m and 6m feature sets with MLR beating SVR with a 3m feature set. All GBRT models outperformed MLR and SVR, achieving best performance when features and target are transformed but without dataset pruning.

FIRST BEST					
Dataset	Naive	ARIMA	Best MLR	Best SVR	Best GBRT
0+	-2.03	-1.70	-0.13	-0.86	0.16
10+	-2.04	-1.70	-0.41	-1.01	0.22
20+	-2.07	-1.73	-0.52	-1.13	0.17
30+	-2.08	-1.74	-0.35	-0.97	0.18
40+	-2.08	-1.74	-0.25	-0.87	0.20
50+	-2.02	-1.69	-0.28	-0.57	0.20

Table 7: best models against naïve, ARIMA, MLR and SVR, considering six-weeks horizon.

FIRST BEST					
Dataset	Naive	ARIMA	Best MLR	Best SVR	Best GBRT
0+	-1.73	-1.50	-0.22	-0.76	0.15
10+	-1.73	-1.50	-0.35	-1.01	0.17
20+	-1.77	-1.54	-0.54	-1.17	0.11
30+	-1.77	-1.53	-0.54	-1.08	0.19
40+	-1.77	-1.53	-0.49	-1.03	0.19
50+	-0.99	-1.49	-0.49	-0.47	0.17

Table 8: best models against naïve, ARIMA, MLR and SVR, considering three-days horizon.

4.5 Features contribution

We also explored how much do specific features contribute to predictions, comparing results obtained for best model to those that only take into account historical values of demand records, annual forecasts or open sales. Best results were obtained with demand history features with an average error of at most 8% greater than from models considering all features, with little variation among those trained for either time horizon. Models considering annual sales forecast (Model AF) had an error of 1.85 times the error of the best model on average, while models based only on future sales (Model FS) had greater error averaging 2.18 times that of the best models.
We conclude the most important feature is demand history, while the rest of the features contribute to enhance results.

FORECAST 6 WEEKS - FIRST BEST				
Dataset	Model	Model AF	Model FS	Model demand
0+	GBRT-9m-wTT-4Y	1.80	2.24	1.07
10+	GBRT-12m-wTT-3Y	2.01	2.41	1.16
20+	GBRT-12m-wTT-2Y	1.80	2.09	1.15
30+	GBRT-9m-wTT-3Y	1.89	2.20	1.07
40+	GBRT-9m-wTT-3Y	1.89	2.30	1.02
50+	GBRT-3m-wTT-2Y	1.73	1.83	0.96
Average		1.85	2.18	1.07

Table 9: comparison of results with feature sub-sets considering six-weeks horizon.

FORECAST 3 DAYS - FIRST BEST				
Dataset	Model	Model AF	Model FS	Model demand
0+	GBRT-12m-wTT-4Y	1.84	2.43	1.07
10+	GBRT-3m-nTT-4Y	1.82	1.80	1.25
20+	GBRT-12m-wTT-4Y	1.83	2.40	1.02
30+	GBRT-12m-wTT-4Y	1.83	2.59	1.07
40+	GBRT-12m-nTT-4Y	1.69	2.05	1.02
50+	GBRT-9m-nTT-4Y	1.77	1.81	1.06
Average		1.80	2.18	1.08

Table 10: comparison of results with feature sub-sets considering three-days horizon.

4.6 R2 FOR BEST MODELS

After performing the experiments, we computed R2 scores to understand how much variance in the forecasted demand is explained by variables taken into account when performing the prediction. When comparing scores obtained for our best models against those from predictions made by logisticians, we found that our models achieve better scores here too by an average of three to six centesimal points.

5 CONCLUSION AND FUTURE WORK

Best models result in an improvement of 10% to 20% over logisticians predictions for both prediction horizons. There is a smaller gap on the three-day prediction horizon, where both predictions are closer to each other. In general, we observe an improvement in results when considering a higher demand history points density. This is also consistent with results regarding features relative importance.
GBRT consistently displays best performance for both forecasting horizons. Regarding feature sets, we observe most models perform best with features computed in a twelve- or nine-months window. When looking for models for six week forecasting horizon, pruning the dataset to a total of three years was optimal, but degraded results for three days horizon.
In the future we would like to enrich existing datasets with time series embeddings as well as products metadata. Time series embeddings should help identify similar timeseries and help make better predictions on products with similar behavior. Products metadata may be used in a similar way, since similar products should have similar demands. Product similarity can be considered from metadata point of view as well as from purchase closeness: items bought together will have similar demands, even though may have different characteristics.

ACKNOWLEDGMENTS

This work was supported by the Slovenian Research Agency and EU H2020 project FACTLOG under grant agreement No 869951.

REFERENCES

[1]- Lee, Hau L., Venkata Padmanabhan, and Seungjin Whang. "Information distortion in a supply chain: the bullwhip effect." Management science 43.4 (1997): 546-558.
[2]- Lee, Hau L., Venkata Padmanabhan, and Seungjin Whang. "Information distortion in a supply chain: the bullwhip effect." Management science 43.4 (1997): 546-558.
[3]- Williams TM (1984). Stock control with sporadic and slow- moving demand. J Opl Res Soc 35: 939–948. Syntetos, Aris A., John E. Boylan, and J. D. Croston. "On the categorization of demand patterns." Journal of the Operational Research Society 56.5 (2005): 495-503.
[4]- Syntetos, Aris A., John E. Boylan, and J. D. Croston. "On the categorization of demand patterns." Journal of the Operational Research Society 56.5 (2005): 495-503.
[5]- Matsumoto, Mitsutaka, and Shingo Komatsu. "Demand forecasting for production planning in remanufacturing." The International Journal of Advanced Manufacturing Technology79.1-4 (2015): 161-175.
[6]- Brühl, Bernhard, et al. "A sales forecast model for the German automobile market based on time series analysis and data mining methods." Industrial Conference on Data Mining. Springer, Berlin, Heidelberg, 2009.
[7]- Spedding, T. A., and K. K. Chan. "Forecasting demand and inventory management using Bayesian time series." Integrated Manufacturing Systems 11.5 (2000): 331-339.
[8]- Liu, Pei, et al. "Application of artificial neural network and SARIMA in portland cement supply chain to forecast demand." 2008 Fourth International Conference on Natural Computation. Vol. 3. IEEE, 2008.
[9]- I Brühl, Bernhard, et al. "A sales forecast model for the German automobile market based on time series analysis and data mining methods." Industrial Conference on Data Mining. Springer, Berlin, Heidelberg, 2009.
[10]- Spedding, T. A., and K. K. Chan. "Forecasting demand and inventory management using Bayesian time series." Integrated Manufacturing Systems 11.5 (2000): 331-339.
[11]- Brühl, Bernhard, et al. "A sales forecast model for the German automobile market based on time series analysis and data mining methods." Industrial Conference on Data Mining. Springer, Berlin, Heidelberg, 2009.
[12]- Dwivedi, Alekh, Maheshwari Niranjan, and Kalicharan Sahu. "A business intelligence technique for forecasting the automobile sales using Adaptive Intelligent Systems (ANFIS and ANN)." International Journal of Computer Applications74.9 (2013).
[13]- Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM, 2016.
[14]- Yeo, In-Kwon and Johnson, Richard (2000). A new family of power transformations to improve normality or symmetry. Biometrika, 87, 954-959.