ICIC Express Letters
Part B: Applications
Volume **14**, Number **1**, January **2023**

 ISSN 2185-2766
pp. **21–28**

BAKERY DEMAND FORECASTING USING XGBOOST AND K-MEANS CLUSTERING

Nathaniel Wikamulia*, Maverick Jonathan and Sani Muhamad Isa

Computer Science Department, BINUS Graduate Program – Master of Computer Science
Bina Nusantara University
JL. K. H. Syahdan No. 9, Kemanggisan, Palmerah, Jakarta 11480, Indonesia
{ maverick.jonathan; sani.m.isa }@binus.ac.id
*Corresponding author: nathaniel.wikamulia@binus.ac.id

Received June 2022; accepted September 2022

Abstract. *This research was conducted to create a machine learning model with XGBoost to perform demand forecasting for a bakery called XYZ Bakery. In addition, this study also compares two approaches in making machine learning models for the bakery business: the first approach is to cluster the dataset first using K-Means to reduce the number of machine learning models created and the second approach is to create machine learning models for each customer. After the experiment, the average accuracy of the model using the first approach was 54.52%, while the second approach resulted in an average accuracy of 86.43%. It can be concluded that in the case of XYZ Bakery, the second approach is better than the first approach, and also XGBoost is proven as an algorithm that can provide good results for demand forecasting due to its ability to surpass previous researches.*

Keywords: Clustering, Demand forecasting, K-Means Clustering, Machine learning, XGBoost

1. **Introduction.** "XYZ Bakery" is a bakery that uses a consignment system to sell its bread, which means the XYZ Bakery will leave its bread to a store and the store only needs to pay for bread that has been sold, while the unsold bread will be returned. This is advantageous for the stores because they do not have to worry about experiencing losses if there is unsold bread. However, this is detrimental for XYZ Bakery because if a lot of bread is returned then the bread can no longer be sold since it has exceeded its expiration date.

To overcome this problem, demand forecasting using machine learning is performed. Demand forecasting is the process of using historical data to predict the quantity of goods to be produced. It is performed to ensure that the goods meet the market's demand, lowering product costs, and reducing the number of unsold products [1,2]. Machine learning is often used to perform demand forecasting because it provides good predictive accuracy [3].

The objective of this research is to create two machine learning (ML) models to perform demand forecasting on the amount of bread that should be produced and to compare the performance of the two ML models that are built with two different approaches. The first approach will cluster the customer dataset and then make predictions, while the second approach (the proposed method of this research) will create an ML model for each customer without clustering. The first approach performs clustering before making predictions because customer behavior in general can be categorized into clusters [4].

There are several studies that are similar to this research that have been done before. Anjum performed demand forecasting and predicted customer transaction times using K-Means and XGBoost. The research resulted in a prediction accuracy of 62% [5]. Tang also

DOI: 10.24507/icicelb.14.01.21

conducted research on demand forecasting using K-Means and XGBoost. The research resulted in a model with an accuracy of 81% [6]. Mouatadid and Adamowski conducted research on demand forecasting using multiple machine learning algorithms and concluded that extreme learning machine is the best method with an accuracy of 81.73% [7]. Tanizaki et al. conducted a demand forecasting research and produced a final accuracy of 85% using Bayesian linear regression [8].

The main contributions of this research are as follows.

- This study tries to compare the method made by previous researches which is creating model based on customer segmentation, with the method proposed by this study, namely by making a separate model for each customer.
- This study tries to produce better prediction accuracy than previous researches.

The results of this study indicate that the proposed model can achieve an average model accuracy of 86.43%, which is significantly higher than previous researches.

The remainder of this paper is organized as follows. Section 2 explains how XGBoost, K-Means, and the evaluation metrics work. Section 3 describes the performed research steps. Section 4 shows the results and analysis of the model. Section 5 shows the conclusion of this research.

2. Literature Review.

2.1. Extreme Gradient Boosting (XGBoost).

XGBoost is a machine learning algorithm that can be used to perform regression or classification by combining several weak learning algorithms in the form of a decision tree [9,10]. It is developed by Chen and Guestrin in 2016 through his paper "XGBoost: A Scalable Tree Boosting System" [11]. XGBoost is an improved version of the gradient boosting algorithm that provides good predictive performance and fast code run times without consuming a lot of memory [12-14].

2.2. K-Means Clustering (K-Means).

K-Means is a machine learning algorithm that is used to cluster data into "k" number of clusters. K-Means works by creating a "k" number of cluster centroids, and then calculating the distance of a data point to all cluster centroids using the Euclidean distance algorithm [15-17]. Each data point will be inserted the closest distance cluster. After that, the position of the centroid will be moved to a new position calculated by the average distance of all data point to the previous centroid. This process will be repeated until the specified number of iterations is reached or the position of the centroid does not change anymore.

2.3. R^2 Score.

R^2 Score is a metric that has a value range from 1.0 to any negative value which is used to check the accuracy of the prediction with the actual data [18], the closer the R^2 Score to 0, the worse the model is. If the value of R^2 Score is minus, it means that the predictions made by the model are not in accordance with the trend and have a high level of variance. The formula for R^2 is as follows:

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{1}$$

2.4. Mean Square Error (MSE).

MSE is a metric that calculates the average between predictions and actual data, the smaller the MSE value means the closer the predicted to the actual data value [19], which means the smaller the MSE value is, the more general (better) the model can accurately predict the target. The formula for MSE is as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2}$$

2.5. **accuracy_score.** accuracy_score is a metric that calculates the accuracy of the model by calculating the percentage of correct predictions divided by the number of test datasets, the closer the accuracy_score value to 100%, the better the model is. The formula for accuracy_score is as follows:

$$accuracy(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples-1}} 1 \ (\hat{y}_i = y_i) \tag{3}$$

3. **Methodology.** The machine learning algorithms that will be used in this research are the Extreme Gradient Boosting (XGBoost) and K-Means Clustering. XGBoost was chosen as the algorithm to perform demand forecasting because in Abbasi et al.'s research, XGBoost produces a good prediction accuracy of 97.21% [20]. In addition, in Sukarsa et al.'s research, XGBoost produces a good prediction accuracy of 97.54% [21]. K-Means Clustering algorithm will be used to cluster the customer dataset. K-Means Clustering will be used as the clustering algorithm in this research because K-Means Clustering is one of the most popular and widely used clustering algorithms [22,23].

This research is divided into six main parts: data gathering, data preprocessing, data clustering or data grouping, model development, model evaluation, and analysis. The workflow of this research can be seen through this image.

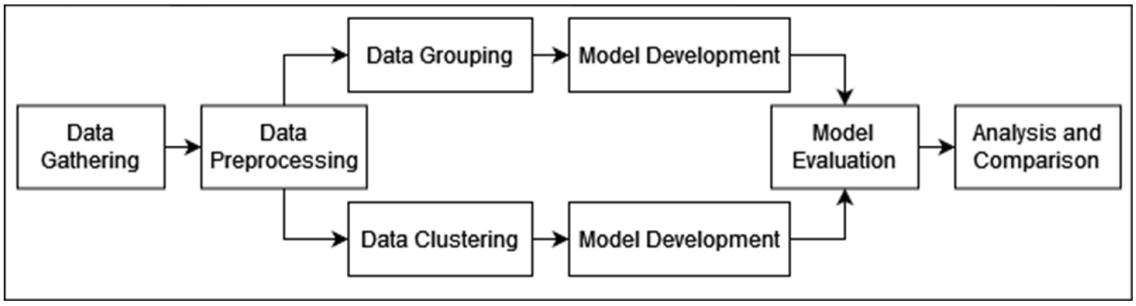

FIGURE 1. Research stages

3.1. **Data gathering.** Based on Figure 1, the first stage in this research is to collect the necessary data needed. The data is provided in the form of an Excel file from the company's database. The data consist of 5 tables, namely: Area, Customer, Inventory, Sales Detail, and Sales Header. The example of the data is shown in Figure 2.

Based on Figure 2, it can be seen that there are still records that have empty value (NaN) and irrelevant columns (customer name, area code) that need to be processed.

CODE	AREACD	SALTYPE	CUSTNAME	ARBAL	INACTIVE
BGR003	BGR	K	BGR003	418000	False
BGR004	BGR	K	BGR004	47500	False
BGR006	BGR	T	BGR006	0	NaN
BGR008	BGR	K	BGR008	478500	NaN
BGR010	BGR	K	BGR010	0	NaN

FIGURE 2. Customer data sample

3.2. **Data preprocessing.** After getting the data, the research continued to the second stage, the data preprocessing stage, which includes converting data from Excel file to Comma Separated Values (CSV) so that it can be read by Python; loading the CSV file into Python using "read_csv" function from the pandas library; deleting null rows or NaN on each table; performing JOIN operation to all data; deleting columns that will not be used as the machine learning features such as primary keys; and selecting which feature should be used based on "plot_importance" function from the XGBoost library. The example of data preprocessing result is as follows.

CODE	AREACD	SALTYPE	CUSTNAME	ARBAL	INACTIVE
BGR003	BGR	0	BGR003	418000	0
BGR004	BGR	0	BGR004	47500	0
BGR012	BGR	1	BGR012	0	0
BGR015	BGR	0	BGR015	827500	0
BGR017	BGR	0	BGR017	0	1

FIGURE 3. Customer data sample after preprocessing

Based on Figure 3, it can be seen that the records have been processed to not contain any empty value (NaN) and some of the value types have been converted to numeric value in order to fit the XGBoost model.

3.3. **Data grouping and data clustering.** In the third stage, the research is divided into two substages: data clustering and data grouping. The data clustering substage is carried out to prepare the dataset for a model that uses clustering. The steps taken at the data clustering stage include running the K-Means Clustering algorithm on the customer data, adding a column to the dataset containing the cluster number belonging to the data, dividing the dataset based on the cluster number, separating the predictor and the predicted column, and dividing the dataset into 80% train dataset and 20% test dataset.

Meanwhile, the data grouping substage is carried out to prepare the dataset for a model that uses an individual customer data. For the data grouping substage, the steps taken include dividing the dataset by customer, separating the predictor and the predicted column and dividing the dataset into 80% train dataset and 20% test dataset.

3.4. **Model development.** In the fourth stage, several machine learning models are created based on the approach used. In the first approach, one machine learning model will use a training dataset from one cluster. The second approach will produce several machine learning models where one model will use a training dataset belonging to one customer.

This research is implemented using Python and several libraries such as Pandas to load and preprocess the data; scikit-learn to normalize datasets; Matplotlib for plotting datasets; SciPy to calculate the Euclidean distance so that the elbow method can find the suitable number of k in K-Means; NumPy to perform data preprocessing and calculation; and XGBoost to create the XGBoost model.

3.5. **Model evaluation.** In the fifth stage, the performance of each machine learning model will be evaluated using three metrics: R^2 Score, MSE, and using the "accuracy_score" function from the scikit-learn library.

3.6. **Analysis and comparison.** In the last stage, after the model evaluations the performance of the model will be analyzed and compared between the data clustering and XGBoost model and data grouping per customer and XGBoost model on the next section.

4. **Result and Discussion.** After conducting the data gathering and data preprocessing stages, only 55 customers remained which had complete data to be used as machine learning models. For the data grouping approach, the transaction data from each customer will be used to generate one machine learning model, so the data grouping approach will produce 55 machine learning models. While the data clustering approach will divide 55 customer data into 5 clusters because based on the elbow method and analysis of the data distribution, the most suitable number of clusters is 5.

From the 19 available features, 5 features were selected to be used for the training process, which are the amount of bread sold (Quantity), the total bread price (GrandTotal), month of the transaction (Month), year of the transaction (Year), and day of the transaction (Day). These features are selected because the "plot_importance" function determined that these features have the highest correlation to the predicted variable.

After the model development stage is finished, the accuracy of each model will be evaluated. Table 1 shows the accuracy of the XGBoost models that uses K-Means.

Table 1. Accuracy table of XGBoost model that uses K-Means

Cluster number	**R^2 Score**	**MSE**	**accuracy_score**
Cluster 1	0.30	3.92	44.03%
Cluster 2	−1.04	0.21	79.07%
Cluster 3	0.23	3.95	40.69%
Cluster 4	0.50	2.19	49.36%
Cluster 5	0.66	1.06	59.45%
Clusters mean	**0.546**	**2.266**	**54.52%**

Based on the data from Table 1, XGBoost using K-Means produces a mean R^2 Score of 0.546, a mean MSE value of 2.266, and a mean "accuracy_score" value of 54.52%. The best R^2 score was obtained by cluster 2 with a value of −1.04, while the worst R^2 score was obtained by cluster 3 with a value of 0.23. The best MSE value was obtained by cluster 2 with a value of 0.21, while the worst MSE value was obtained by cluster 3 with a value of 3.95. The best "accuracy_score" value was obtained by cluster 2 with an accuracy of 79.07%, while the worst "accuracy_score" value was obtained by cluster 3 with a value of 40.69%. From the comparison of the three metrics, the best cluster is cluster 2 because cluster 2 gets the best score in R^2 Score, MSE and "accuracy_score"; while the worst cluster is cluster 3 because cluster 3 gets the worst score in R^2 Score, MSE and "accuracy_score". Table 2 shows the accuracy of XGBoost model grouped by customer.

Based on the data from Table 2, XGBoost model that uses data grouping approach produces a mean of 86.43%, a median of 86.21%, and a mode of 100%. The three groups that have the highest frequency are group 2 with 13 models, group 4 with 11 models, and

Table 2. Accuracy table of XGBoost model grouped by customer

Group	**accuracy_score**	**Frequency**
Group 1	70.42% – 75.35%	7
Group 2	75.36% – 80.28%	13
Group 3	80.29% – 85.21%	5
Group 4	85.22% – 90.14%	11
Group 5	90.15% – 95.07%	6
Group 6	95.08% – 99.99%	3
Group 7	100%	10

group 7 with 10 models. Although the group that has the highest frequency is in group 2 and group 4, the most frequent value comes from group 7, which is 100% as many as 10 model. The 10 out of 55 models that have an accuracy_score value of 100% indicate that 18% of the models are overfit. This happens because the number of training datasets for the 10 models is still lacking where the average amount of data used for the ten overfit models is 99 data, while the average amount of data used for the non-overfit models is 301 data.

After comparing the accuracy of the two approaches, it can be seen that the first approach that uses XGBoost and K-Means as done by Anjum [5] and Tang [6] produces an average accuracy of 54.52%. In contrast, the second approach, which creates an XGBoost model for each customer, produces an average model accuracy of 86.43%. The accuracy of the first approach is not as good as the second approach because the purchasing pattern from each customer in a cluster is not the same. The following is a comparison chart of purchasing patterns from several customers in the same cluster.

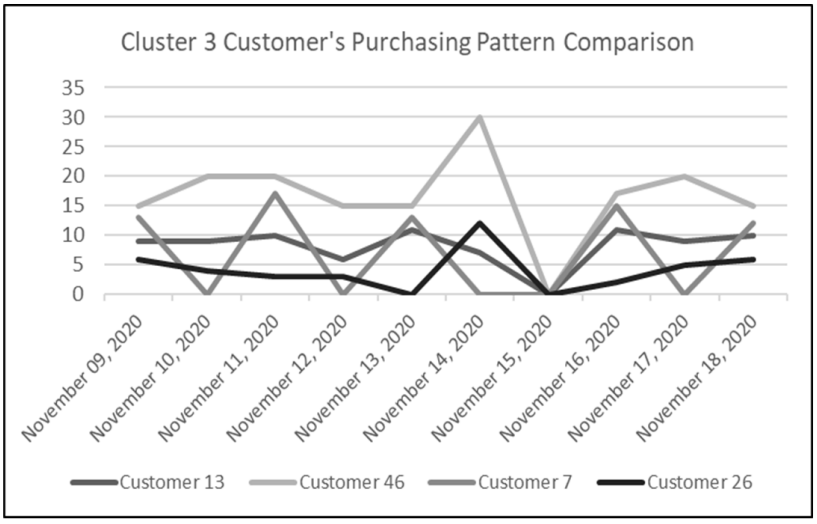

FIGURE 4. Purchasing pattern comparison of 4 XYZ Bakery customers from cluster 3

Figure 4 shows the purchasing pattern of four customers who were randomly selected from cluster 3. Each line displays the amount of bread purchased which have been reduced by the amount of bread returned. The purchasing patterns of the four customers are different from each other. For example, one customer makes a purchase every two days, while others purchase bread every day. The amount of bread purchased by the four customers also varied, customer 13 bought around 6-13 loaves of bread; customer 46 bought around 15-30 loaves of bread; customer 7 buys around 12-17 loaves of bread; and customer 26 buys around 3-12 loaves of bread.

Based on Table 1, Table 2, and Figure 4, it can be seen that for XYZ Bakery's case, the approach that uses K-Means and XGBoost is not suitable to be implemented because the purchasing pattern between each customer is different, so it cannot be made into a cluster as in Anjum's [5] and Tang's [6] research. Therefore, the appropriate approach for the XYZ Bakery case is to provide customized treatment for each customer by creating a model that uses data grouping per customer, where each customer is given a separate XGBoost model.

In addition, when compared with the accuracy in previous researches, the method proposed by this study produces better accuracy. This can be seen from Table 3.

Based on Table 3 it can be seen that the K-Means and XGBoost method resulted in a final accuracy of 54.52% which means it neither matches nor surpasses the performance of the previous research, and meanwhile the proposed method in this study resulted in a

Table 3. Method accuracy comparison

Method	Accuracy
Anjum	62%
Tang	81%
Mouatadid and Adamowski	81.73%
Tanizaki et al.	85%
K-Means and XGBoost (Based on Anjum's and Tang's method)	54.52%
Proposed Method (One XGBoost model for each customer)	86.43%

final accuracy of 86.43%, which means that the proposed method succeeded in surpassing the performance of the previous researches and the predictions produced by this research are more accurate than previous researches.

5. **Conclusion.** The first approach that uses XGBoost and K-Means produces an average model accuracy of 54.52%. In comparison, the second approach that makes an XGBoost model for each customer produces an average model accuracy of 86.43%. Based on the percentage change formula, the second approach is 58.52% better than the first approach because XYZ Bakery's customers have different buying patterns, which means customer cannot be divided into clusters. In addition, the method proposed by this research, namely by creating a model for each customer, has successfully surpassed the accuracy of the methods proposed in previous studies. Although the second approach has better average accuracy than the first, there are still problems with the second approach. One such problem is overfitting due to lack of training dataset. There are improvements that can be made in future research, such as increasing the average model accuracy and comparing XGBoost's performance with other machine learning algorithms.

REFERENCES

[1] J. J. Bergman, J. S. Noble, R. G. McGarvey and R. L. Bradley, A Bayesian approach to demand forecasting for new equipment programs, *Robot. Comput. Integr. Manuf.*, vol.47, pp.17-21, doi: 10.1016/j.rcim.2016.12.010, 2017.

[2] J.-H. Böse et al., Probabilistic demand forecasting at scale, *Proc. of VLDB Endow.*, vol.10, no.12, pp.1694-1705, doi: 10.14778/3137765.3137775, 2017.

[3] K. Kaya, Y. Yılmaz, Y. Yaslan, Ş. G. Öğüdücü and F. Çıngı, Demand forecasting model using hotel clustering findings for hospitality industry, *Inf. Process. Manag.*, vol.59, no.1, 102816, doi: 10.1016/j.ipm.2021.102816, 2022.

[4] A. Candelieri, Clustering and support vector regression for water demand forecasting and anomaly detection, *Water (Switzerland)*, vol.9, no.3, 224, doi: 10.3390/w9030224, 2017.

[5] A. M. Anjum, *Customer Segmentation Using RFM Analysis*, B.Sc. Thesis, Daffodil International University, 2020.

[6] P. Tang, Telecom customer churn prediction model combining K-means and XGBoost algorithm, *Proc. of 2020 5th Int. Conf. Mech. Control Comput. Eng. (ICMCCE 2020)*, pp.1128-1131, doi: 10.1109/ICMCCE51767.2020.00248, 2020.

[7] S. Mouatadid and J. Adamowski, Using extreme learning machines for short-term urban water demand forecasting, *Urban Water J.*, vol.14, no.6, pp.630-638, doi: 10.1080/1573062X.2016.1236133, 2017.

[8] T. Tanizaki, T. Hoshino, T. Shimmura and T. Takenaka, Demand forecasting in restaurants using machine learning and statistical analysis, *Procedia CIRP*, vol.79, pp.679-683, doi: 10.1016/j.procir.2019.02.042, 2019.

[9] V. Morde, *XGBoost Algorithm: Long May She Reign!*, 2019, https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d, Accessed on Nov. 21, 2021.

[10] H. Lv et al., iRice-MS: An integrated XGBoost model for detecting multitype post-translational modification sites in rice, *Brief. Bioinform.*, vol.23, no.1, pp.1-13, doi: 10.1093/bib/bbab486, 2021.

[11] T. Chen and C. Guestrin, XGBoost: A scalable tree boosting system, *Proc. of ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, pp.785-794, doi: 10.1145/2939672.2939785, 2016.

[12] S. Gupta, L. Goel, A. Singh, A. K. Agarwal and R. K. Singh, TOXGB: Teamwork optimization based XGBoost model for early identification of post-traumatic stress disorder, *Cogn. Neurodyn.*, vol.1, doi: 10.1007/s11571-021-09771-1, 2022.

[13] C. Wang, C. Deng and S. Wang, Imbalance-XGBoost: Leveraging weighted and focal losses for binary label-imbalanced classification with XGBoost, *Pattern Recognit. Lett.*, vol.136, pp.190-197, doi: 10.1016/j.patrec.2020.05.035, 2020.

[14] J. Brownlee, *A Gentle Introduction to XGBoost for Applied Machine Learning*, 2016, https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/, Accessed on Nov. 30, 2021.

[15] Y. Abo-Elnaga and S. Nasr, K-Means cluster interactive algorithm-based evolutionary approach for solving bilevel multi-objective programming problems, *Alexandria Eng. J.*, vol.61, no.1, pp.811-827, doi: 10.1016/j.aej.2021.04.098, 2022.

[16] A. Shirazy, A. Hezarkhani, A. Shirazi, S. Khakmardan and R. Rooki, K-Means clustering and general regression neural network methods for copper mineralization probability in Char-Farsakh, Iran, *Türkiye Jeol. Bülteni/Geol. Bull. Turkey*, vol.64, pp.79-92, doi: 10.25288/tjb.1010636, 2021.

[17] D. Cohen, T. Lee and D. Sklar, *Precalculus: A Problems-Oriented Approach*, Cengage Learning, 2004.

[18] K. Rao, r^2 *or* R^2 *– When to Use What*, 2020, https://towardsdatascience.com/r^2-or-r^2-when-to-use-what-4968eee68ed3, Accessed on Dec. 27, 2021.

[19] T. Danka, *Where Does the Mean Squared Error Come from?*, 2021, https://towardsdatascience.com/tagged/mean-squared-error?p=710e0bf6bcf8, Accessed on Dec. 27, 2021.

[20] R. A. Abbasi, N. Javaid, M. N. J. Ghuman, Z. A. Khan, S. Ur Rehman and Amanullah, Short term load forecasting using XGBoost, in *Artificial Intelligence and Network Applications. WAINA 2019. Advances in Intelligent Systems and Computing*, L. Barolli, M. Takizawa, F. Xhafa and T. Enokido (eds.), Pakistan, Springer International Publishing, 2019.

[21] I. M. Sukarsa, N. N. Pandika Pinata, N. Kadek Dwi Rusjayanthi and N. W. Wisswani, Estimation of gourami supplies using gradient boosting decision tree method of XGBoost, *TEM J.*, vol.10, no.1, pp.144-151, doi: 10.18421/TEM101-17, 2021.

[22] A. Dubey and A. Choubey, A systematic review on K-Means clustering techniques, *Int. J. Sci. Res. Eng. Technol.*, vol.6, no.6, pp.624-627, 2017.

[23] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. Piatko, R. Silverman and A. Y. Wu, The analysis of a simple K-Means clustering algorithm, *Proc. of the 16th Annual Symposium on Computational Geometry (SCG'00)*, pp.100-109, doi: 10.1145/336154.336189, 2000.