Predicting Pregnant Shoppers Based on Purchase History Using Deep Convolutional Neural Networks

Nikhil Angad Bakshi, Prithviraj Reddy Kolan, and Bibek Behera
Sears Holdings Corporation, Pune, MH, India
Email: Nikhil.Bakshi@searshc.com, {Prithviraj.Kolan, Bibek.Behera}@searshc.com

Naveen Kaushik
Sears Holdings Corporation, Bengaluru, KA, India
Email: Naveen.Kaushik@searshc.com

Ansari Mohammed Ismail
Sears Holdings Corporation, Seattle, WA, USA
Email: Ansari.MohammedIsmail@searshc.com

Abstract—**Predicting pregnant shoppers based on their transaction history and purchasing trends is a challenging problem because of data sparsity and imbalance. Not only are the instances, i.e., the ratio of pregnant versus non-pregnant shoppers, skewed but also is the proportion of the products that reveal pregnancy status. The problem of class imbalance has been solved by taking an equal number of positive and negative examples for training and testing while deployment has been done on the entire dataset yielding results that were congruent with the real-world. In this paper, we use a novel approach that uses deep Convolutional Neural Networks (CNNs) to handle one dimensional data. The proposed solution overcomes the above mentioned challenges and proves that two dimensional CNNs outperform a baseline LightGBM (gradient boosting framework that uses tree based learning algorithms) model on two different datasets - the dataset based on twenty one *hot products* and the dataset based on all products by subcategory. The CNN model reached an F1 score of 0.69 on the test set.**

Index Terms—**convolutional neural networks, pregnancy detection, retail prediction, deep learning, pregnant shopper behavioral trends**

I. INTRODUCTION

This work was done as part of a marketing campaign targeting pregnant shoppers for Sears Holdings Corporation. The underlying assumption is that buying patterns among pregnant shoppers is different from others. This research has been carried out pursuant to our privacy policy and hence transactions have been reported as a transaction index and our predictions have been reported as percentages of the total population rather than absolute numbers.
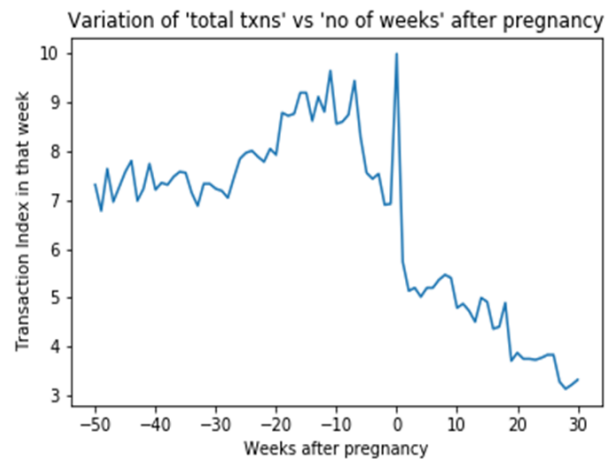

Figure 1. On the x-axis the zeroth week indicates childbirth. There is a clear decrease in sales at this point. Transaction index is the scaled count of transactions in each week

The key observation while analyzing this problem was that the shopping pattern of customers changes drastically at three key points in their life: graduation, marriage and childbirth. This change usually implies a major change in the type of products a customer shops for and hence most likely a change of retailer too. Our analysis (Fig. 1) confirms that there is huge potential to be tapped in pregnant shoppers. It is possible that they are moving to online shopping or other offline stores. The motivation to solve this problem is twofold, firstly, we increase revenues obtained from pregnancy period consumables and, secondly, if we are able to retain those shoppers during their pregnancy period, we increase the likelihood they turn to us for their post-natal needs as well translating to their long term retention.

In this paper we use two dimensional images to represent the shopping history or purchase pattern of each customer. One axis of the image is on time and the other

is on the product category. We found that the 2D CNN approach worked superior to a 1D CNN approach.

There were three reasons behind taking a 2D image approach, firstly, we leverage the pattern extraction capabilities of CNNs to identify complicated combinations of products by using different sized filters as a completely diverse combination of products could be brought into the feature space that were not plausible in a one dimensional approach using RNNs (recurrent neural networks) which would be a more conventional approach to transactional data, secondly, we can still capture the variation in sales with respect to time, thirdly, it often happens that shoppers are not regular and instead shop in larger quantities but less frequently, this leads to many null vectors being fed to the model in an RNN approach and this problem is solved using a 2D image approach with convolutional filters as patterns can be detected anywhere in the image without information fading with time.

The rest of the paper is organized as follows. In Section 2 we touch upon some of the relevant work in this area. After this, we describe the datasets we built in Section 3. We then move on to the analysis done on the data in Section 4. After discussing the analysis, we make a detailed description of our neural network design in Section 5. The experiments conducted are described in Section 6. Hardware we used in the experiments has been described in Section 7 followed by results and conclusion in Section 8 and Section 9, respectively. Finally, plans for future work have been covered in Section 10.

II. Related Work

This problem has previously been tackled by Target in the 2000s with news-worthy success [1], however the methodologies/algorithms used by Target were never made public. Since then the face of predictive algorithms has changed. With the emergence of deep learning and the availability of more data we no longer need to heuristically identify the exact products that indicate the pregnancy status of a shopper.

Cumby *et al.* [1] attempt to model shoppers' behaviour as a classification problem on customer product pairs to predict the shopping list on each trip using a shopping vector of handmade features for previous trips to make predictions similar to the POS tagging approach in the field of NLP. Salehinejad and Rahnamayan [2] use the more standard approach of Recurrent Neural Networks (RNNs) to time series data such as in this paper to model every customer's value to the business which can then be used to for further marketing campaigns.

In this paper we present one of the first, if not the first, published account of, in a research context, the approach and solution to this particular problem in the retail world.

III. Dataset

Out of the millions of shoppers at Sears some of them have declared their children's birthdays/due dates to us and these are the primary set of positive training examples we use in the model. While the primary set of positive training examples were mothers who had registered with us, we did not filter our shoppers' data on the basis of age or gender as the shopping patterns exhibited by a pregnant mother could reasonably be emulated by anyone in her household.

The negative set of training examples were those members with transactions in any category except those unequivocally related to prenatal shopping such as prenatal vitamins, maternity clothes or other such products.

All transactions used in training and testing date back no earlier than 2013. We have trained on data from all months in a year to avoid trends based on seasonality causing errors.

The positive and negative examples were taken in equal measure in the training set. Pursuant to our privacy policy we cannot disclose the exact numbers of pregnant and non-pregnant examples used to train and validate the model.

There were two structures to the dataset:

- **Heuristically found hot products**

Based on the shopping trends of the pregnant shoppers relative to their due date we shortlisted a set of twenty-one product groups that contained important information to predict the pregnancy status of a shopper either by virtue of them being associated with pregnant mothers, avoided by pregnant mothers (alcohol, cigarettes, etc.) or being general consumption products (chocolates, chips, etc.) (see Table I). They are:

TABLE I.   Twenty-One Shortlisted Hot Products

Soft Drinks	Beer
Prenatal vitamins	Tampons
Chocolate	Pumpkin seeds
Contraceptives	Baby products
Rootbeer	Cigarettes
Wine	Infant
Home Pregnancy Test	Sanitizers
Chips	Coffee related
Ice cream	Maternity things
Cotton balls	Newborn
Non wine	-

For the Keras-based CNN model the data of these hot products was arranged in a 2-D image with products on the *y*-axis. On the *x*-axis was the month-wise quantity of sale as well as one entry for the cumulative sales over all the months. It was experimentally found that 3 months history performed the best. Therefore the image for a given shopper at a given time was a 21x4 image. Insights into the trends of these 21 product groups can be found in Section 4. This was done because the convolutional filters could

---

figure out a gradual change of buying patterns in the last 3 months as well as find relevant weights for combination of products in the matrix that could have hidden patterns for pregnancy. The underlying idea was that across columns there is a temporal variation and across rows there are product patterns. The data was passed as an unrolled vector in the case of the LightGBM baseline classifier. (Fig. 2)
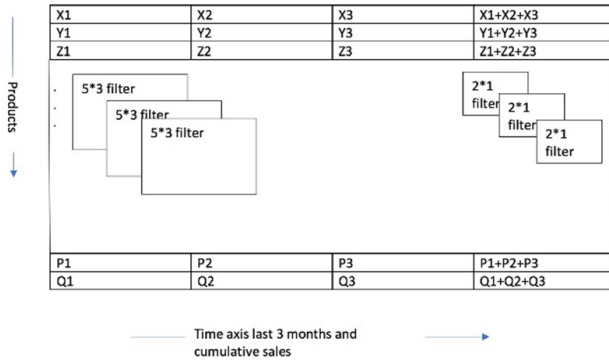

Figure 2. Image representation of a feature vector. The filters act along different regions of the image some extracting product patterns and others extracting temporal patterns.

- **Unfiltered list of products on sub-category level**

As the number of unique product subcategories were in the order of tens of thousands in this dataset structure, at a given time for a given shopper a vector of length equal to the number of subcategories was created. The entries in this vector was cumulative sales in that subcategory over the previous 3 months by that shopper. The vector was then reshaped into a perfect square image after zero padding. Perfect square image gives us more flexibility in experimenting with different sized filters. The primary objective in this dataset structure was to find patterns between products as 3 months history was already known to perform the best. This data was used for the PyTorch-based CNN model. This dataset was created to remove the human component of deciding what products are important, along with that one must appreciate that given shopping data is often extremely sparse and shoppers are not necessarily regular the majority of the month-wise shopping vectors created for a given customer for a given number of past months would be null. This is the key challenge if one were to take an RNN approach and the primary reason we did not make separate columns for separate months of shopping in each 2D image.

IV. ANALYSIS

In the figures of this section (Fig. 3 – Fig. 7) the *x*-axis represents number of months after delivery (negative value implies number of months before delivery). Zero indicates the month of delivery, which is the due date/baby's birthday and the interval [−9, 0] is the pregnancy period.

The graphs in this section represent transactions since 2013 of all shoppers for whom due date information is available. The transactions in the various product categories were tallied on a monthly basis with respect to the due date and then scaled between 0 and 10 so as not to reveal confidential information. The products have been clubbed in the same graphs when the scale of transactions is comparable for clarity. The other 11 products plots do not have easily discernible trends and have not been included in this section. The products were selected either because they are general consumption products or because they are recommended/not recommended for pregnant women.
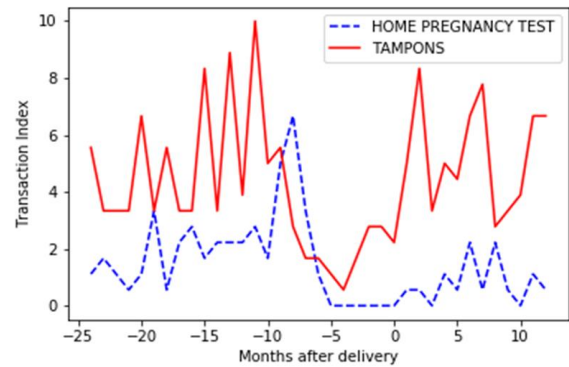

Figure 3. 1) The solid red line shows a dip in sales of tampons during the period of pregnancy which is congruent with what we expect. 2) The dashed blue line has a spike at -8 with high values at -9 and -7 testifying that home pregnancy tests are purchased primarily in the first trimester. Additionally we notice a certain datum level of purchase in -10 and earlier, this could indicate the shopper is trying to get pregnant.
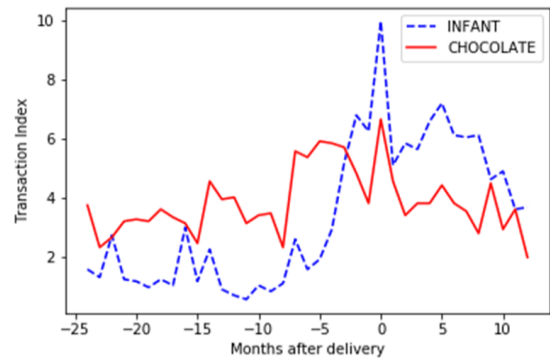

Figure 4. 1) The solid red line shows that chocolate sales peak around delivery with an additional high in the second trimester. 2) The dashed blue plot shows that sales of infant related products starts to climb in the third trimester and peaks around delivery.
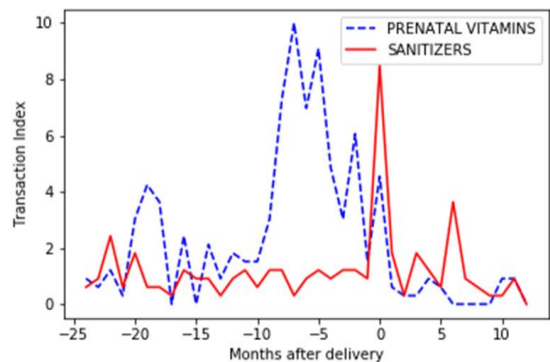

Figure 5. 1) The solid red line has a peak at 0 indicating that sanitizers are purchased in high amounts at the time of delivery. 2) The dashed blue line shows prenatal vitamin sales peak in the first trimester and taper down until delivery.

Figure 6. 1) The solid red line shows baby products sales begin to climb in the second trimester and make a sudden peak at the time of delivery. 2) The dashed blue line shows soft drinks sales peak beyond the usual around the time of delivery.
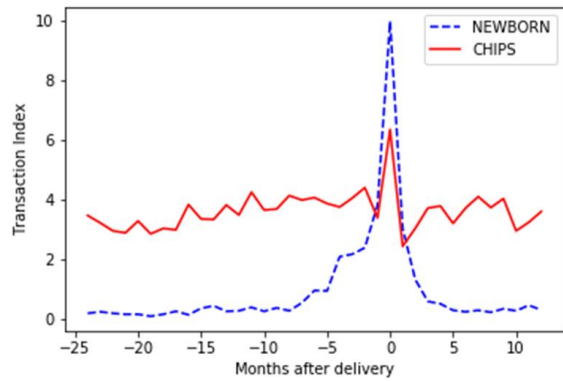

Figure 7. 1) The solid red line shows that chips sales peak around delivery to nearly twice the usual value. 2) The dashed blue line shows that newborn related products begin to climb in the second trimester from virtually nil to a huge peak at the time of delivery.

V. MODEL DESCRIPTION

The first dataset was used to train 2D convolutional neural nets (convnets) on Keras [3] platform with Tensorflow [4] backend as well as on the baseline LightGBM model. For the baseline the training data was fed in as a 1D vector.

The second dataset was used to train the PyTorch [5] based 2D convnet model with GPU support to accelerate training speed. We initially had square filters on the convent layers [6], but later on moved to filters with sizes that were in the sequence of Fibonacci numbers. Our experiments showed that the rectangular filters performed better. Every convnet layer is followed by a LeakyReLU [7] activation layer as it prevents the problem of dead nodes faced by ReLU activated layers when the inactivated values are negative.

The cascade of convnet layers was followed by max pool layers and 2 fully connected (FC) layers decreasing from 64 neurons in the first FC layer to 2 neurons in the last FC layer. The design of the Keras-based model has been shown in Figure 8. The reason for having a cascade of FC layers instead of a single FC layer is that the transition of information between the flattened vector and

64 neuron FC layers is smoother compared to that between the flattened vector and 2 neuron FC directly.
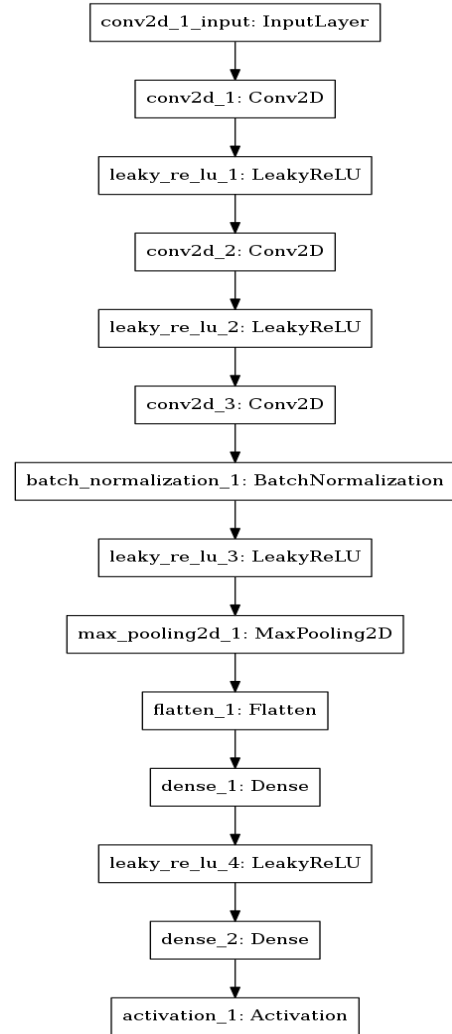

Figure 8. Modular representation of the neural net we used for training with 21 hot prods dataset in the Keras model

In the PyTorch model we have used 2 convnet layers. The first layer consists of 4 square filters of size 15 followed by LeakyReLU and MaxPool while the next layer consists of 3 square filters of size 11 followed by ReLU and AvgPool (Average Pool). The first convnet layer is followed by Max Pool as the input size is very large and sparse for this model (using the second dataset) while the second convnet layer is followed by AvgPool. While experimenting we found that if we apply AvgPool first and MaxPool second, we obtain a lower f1-score.

VI. EXPERIMENTS

We conducted three experiments as described in Table II. Since the negative cases are enormous, the overall F1 score will distort the real performance of the model. Given the purpose is to predict the pregnant shoppers accurately we prefer the F1 score of only the positive class.

TABLE II. EXPERIMENTS WITH DIFFERENT MODEL AND DATASET

Model	Dataset	F1 Score (positive class)
Light GBM	21 hot prod	0.51
Keras	21 hot prod	0.56
PyTorch	Full vector	0.69

We also made a number of experiments for the torch based model for selecting the best configuration as shown in Table III. The table shows that by altering the size of convolutional layer, we can improve the accuracy of model.

TABLE III. EXPERIMENTS WITH DIFFERENT ARCHITECTURE IN PYTORCH. CL MEANS CONVOLUTION LAYER. SINGLE VALUE IMPLIES IT'S A SQUARE FILTER WHILE A TUPLE MEANS A RECTANGULAR FILTER

| Model	CL #1	CL #2	CL #3	F1 Score
1	11	26	NA	0.624
2	5	3	3	0.608
3	21	13	8	0.654
4	11	26	NA	0.644
5	(21,13)	(13,5)	4	0.667
6	15	11	NA	0.69

VII. EXPERIMENTAL SETUP

All experiments were carried on google cloud platform (GCP). The LightGBM [8] baseline model and the 21 hot products Keras-based CNN model including all their data processing and training was run on a single machine with 6 CPUs and 30 GB of RAM.

The neural network using PyTorch was trained on a VM with 16 CPUs, 60 GB Ram and 100 GB hard disk. This configuration was additionally powered with 2 NVIDIA Tesla K80 GPUs.

VIII. RESULTS

The accuracy has been reported in Table II. The various experiments to figure out the right configuration for the CNN is shown in Table III.
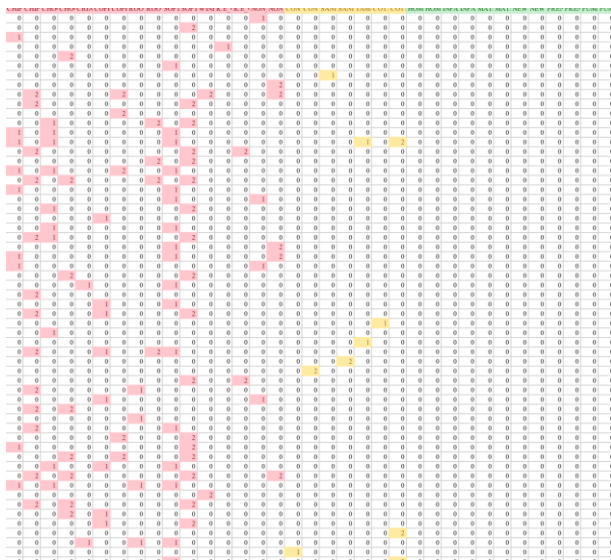
(a) A visual representation of the shopping vectors of shoppers known to be pregnant but have been misclassified by the CNN model
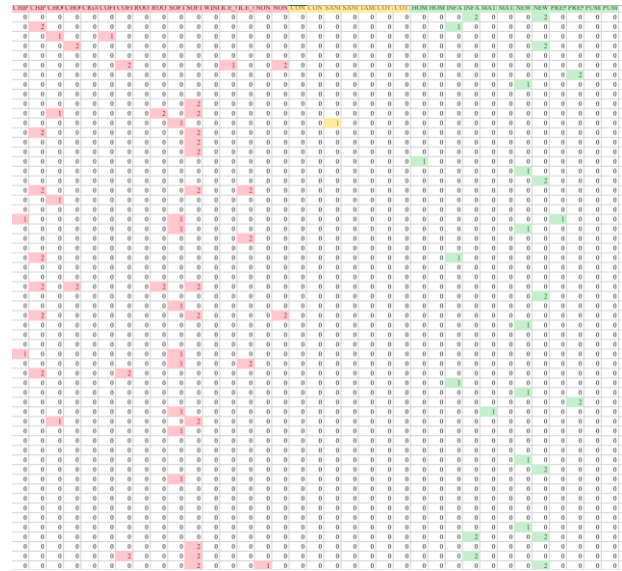
(b) A visual representation of the shopping vectors of shoppers known to be pregnant which have been correctly classified by the CNN model.

Figure 9. Red indicates non-zero entries for chips, chocolates, cigarettes, root beer, wine, non-wine, coffee, soft drinks and ice-cream. Yellow indicates non-zero entries for contraceptives, cotton balls, sanitizers and tampons. Green indicates non-zero entries for home pregnancy tests, maternity items, prenatal vitamins, pumpkin seeds and newborn/infant related items. As is evident, there are a significant number of green items in Fig. 9b while red and yellow items are more common in Fig. 9a. This indicates that the model is performing well and the misclassified shoppers are justified.

Of all shoppers that recorded a transaction with us in the 21-hot products in the 3 month period prior to 22nd January 2018, the model predicted nearly 20% of them as potentially pregnant. This is an acceptable result because out of the total shoppers from the 3 month period in question, only about 26% of them had a transaction in the 21 hot products, therefore from the entire population of shoppers the number tagged as pregnant is 5%.

This is congruent with what one would expect in the real-world. We have forwarded the details of these shoppers to our marketing team where these members will be targeted in a campaign allowing us to assess our performance in the real world.

On the same data the baseline LightGBM model predicted nearly 35% shoppers as pregnant. The plausibility that such a large number of the shoppers are pregnant is considerably lower and it is evident that the baseline has been outperformed.

The key take-away at this point is that the 2D image approach using CNNs has outperformed the 1D vector approach to the data modelling even with an advanced library like LightGBM.

Upon looking deeper into the results of the model (Fig. 9) we see that the misclassified shoppers are justified and the model has detected the trends sufficiently well to generalize to the global set of shoppers.

Next we look at the PyTorch model, the key take away at the improved performance of this model is that the feature engineering done heuristically was insufficient to capture all the trends in the data. Given the size of representation of the customers in the dataset for this model, the model was not scaled up to make a prediction

on the test period and hence measure its selectivity however we are working to move the improved model into production and will address the general performance of this model in future work. Secondly we can infer that the loss of temporal information in the 2D image for this model did not adversely affect its performance we would argue because of the inherent sparseness of transactional data. This is an important finding with respect to our future work as it is a good solution to tackling the problem of null vectors making the data unwieldy in an RNN approach or while using the first dataset structure defined in Section 3.

IX. CONCLUSION

In this paper we have shown the success of a CNN approach to time series data over a baseline model given the same information in a 1D format. We have built an architecture and defined a structure to the dataset that retains temporal information while giving the model the opportunity to learn patterns between products even if they haven't been purchased together.

Next we showed that even with the loss of temporal information from this 2D image whilst taking shopping data at a more general level we were able to further improve the performance of the model.

Additionally the results of the model showed a satisfactory separation of the shoppers into the two classes. In absence of large-scale tagged data, the visual substantiation of the capabilities of the model were sufficient to put the model into production and predict on real-world untagged data.

This paper is one of the forerunners to present a solution to the problem of pregnancy prediction on the basis of shopping history.

X. FUTURE WORK

This paper primarily deals with predicting the pregnancy status of a shopper as a binary variable, however as we have seen in Section 4 there are deeper trends that could be leveraged to predict to finer grain at trimester/month level. This can be taken up in future work.

Recurrent Neural Networks (RNNs) are the usual choice for data which naturally comes ordered in time. A comparative study using an RNN approach with the results of this paper can also be taken up in the future to prove the problems arising with sparsity of retail data.

A deeper analysis and scaling up of the best performing model presented in this paper will also be taken up in future work.

REFERENCES

[1] C. Cumby, A. Fano, R. Ghani, and M. Krema, "Predicting customer shopping lists from point-of-sale purchase data," in *Proc. Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 2004, pp. 402–409.

[2] H. Salehinejad and S. Rahnamayan, "Customer shopping pattern prediction: A recurrent neural network approach," in *Proc. IEEE Symposium Series on Computational Intelligence*, 2016, pp. 1–6.

[3] F. Chollet, *et al.* (2015). Keras: Deep learning library for theano and tensorflow. [Online]. Available: https://keras.io/k

[4] M. Abadi, *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," 2015.

[5] A. Paszke, S. Gross, S. Chintala, and G. Chanan, Pytorch, 2017.

[6] H. Zeng, M. D. Edwards, G. Liu, and D. K. Gifford, "Convolutional neural network architectures for predicting dnaprotein binding," *Bioinformatics*, vol. 32, no. 12, pp. i121–i127, 2016.

[7] P. Ramachandran, B. Zoph, and Q. Le, "Searching for activation functions," 2017.

[8] G. Ke, *et al.*, "Lightgbm: A highly efficient gradient boosting decision tree," *Advances in Neural Information Processing Systems*, pp. 3149-3157, 2017.

Nikhil Angad Bakshi, born in India, obtained a Bachelor's in Technology from Indian Institute of Kharagpur, West Bengal, India with a major in mechanical engineering, minor in computer science and engineering and a micro-specialization in wireless embedded systems.

He is currently a Business Analyst at Sears Holdings Corporation at their offices in Pune, India. He has previously worked as a KITES Summer Intern at ITC Ltd. and as a Research Intern at Navstik Labs. His notable publications in the past are:

Nikhil Angad Bakshi (March 2018). Model Reference Adaptive Control of Quadrotor UAVs: A Neural Network Perspective, Adaptive Robust Control Systems Anh Tuan Le, IntechOpen, DOI: 10.5772/intechopen.71487.

Mayank Singh, *et al.* (June 2017). Citation sentence reuse behavior of scientists: A case study on massive bibliographic text dataset of computer science, ACM/IEEE Joint Conference on Digital Libraries (JCDL), DOI: 10.1109/JCDL.2017.7991589

Nikhil Angad Bakshi, Raghu Ramachandran (Jan. 2016). Indirect model reference adaptive control of quadrotor UAVs using neural networks, 10th International Conference on Intelligent Systems and Control (ISCO), DOI: 10.1109/ISCO.2016.7727123

His research interests are in the fields of deep learning, robotics, natural language processing and predictive analytics.

Prithviraj Reddy Kolan, born in India, obtained a Bachelor's degree in Technology from Indian Institute of Technology Kharagpur, WB, India in Computer Science and Engineering.

He is currently a Senior Business Analyst at Sears Holdings Corporation at their offices in Pune, India. He has previously worked as Software Developer Intern at Samsung R&D Institute Bangalore, India.

His research interests are in the fields of Deep Learning, Machine Learning and Artificial Intelligence.

Bibek Behera, born in India, obtained a Master's degree in Technology from Indian Institute of Technology, Bombay in Computer Science and a Bachelor's degree in Technology from Indian Institute of Technology, Bombay in Energy Engineering.

He is currently working as Algorithm engineer II in Alibaba China co. ltd. in Hangzhou, China. Previously he worked in Sears Holdings India, Affine Analytics, Magictiger with a total experience of 3+ years.

His previous publications are as follows:

Behera, Bibek, *et al.* "Distributed Vector Representation Of Shopping Items, The Customer And Shopping Cart To Build A Three Fold Recommendation System," arXiv preprint arXiv:1705.06338 (2017).

Behera, B., Lalwani, A., & Awate, A. (2014). Using Webcam to Enhance Fingerprint Recognition. In Articulated Motion and Deformable Objects (pp. 51-60). Springer International Publishing.

Behera, B., & Bhattacharyya, P. (2013). Automated Grammar Correction Using Hierarchical Phrase-Based Statistical Machine Translation. In Proceedings of IJCNLP (pp. 937-941).

Naveen Kaushik, born in India, obtained a Bachelor's in Engineering from M V J College of Engineering, Bangalore, Karnataka, India with a major in Computer Science.

He will be joining Monash University, Melbourne, Victoria, Australia in July 2018 to pursue a course of Master of Data Science. He has previously worked as a Senior Technical Associate at Sears Holdings Corporation at their offices in Bangalore, India, as a Data Scientist for Affine Analytics at Bangalore, India and as a Senior Software Engineer for Impetus Infotech (India) Pvt. Ltd. at Bangalore, India.

His research interests are in the fields of deep learning, reinforcement learning, natural language processing and predictive analytics.

Ansari M. Ismail is a machine learning engineer with 15 years of experience. He was part of Microsoft research and Azure ML. His current focus is on deep learning based reinforcement algorithms and NLP.