Zero to One: Sales Prospecting with Augmented Recommendation[†]

Saiquan Hu[‡] Juanjuan Zhang[§] Yuting Zhu[¶]

February 15, 2022

Abstract

Helping new salespeople succeed is critical in sales force management. We develop a deep learning based recommender system to help new salespeople recognize suitable customers, leveraging historical sales records of experienced salespeople. One challenge is how to learn from experienced salespeople's own failures, which are prevalent but often do not show up in sales records. We develop a parsimonious model to capture these "missing by choice" sales records and incorporate the model into a neural network to form an augmented, deep learning based recommender system. We validate our method using sales force transaction data from a large insurance company. Our method outperforms common benchmarks in prediction accuracy and recommendation quality, while being simple, interpretable, and flexible. We demonstrate the value of our method in improving sales force productivity.

Keywords: sales force management, deep learning, recommender system, neural network, selection bias.

[†]Author names are listed alphabetically. The authors thank faculty and students at the MIT Sloan School of Management for their incredible support. The authors also received helpful comments from Alessandro Bonatti, Doug Chung, Kinshuk Jerath, Jinhong Xie; from Shankar-Spiegel Dissertation Proposal Award reviewers; from seminar participants at the City University of Hong Kong, Massachusetts Institute of Technology, National University of Singapore, Shanghai Jiao Tong University, University of Chicago, University of Guelph; and from audiences at the 2021 Artificial Intelligence in Management Conference, 2021 Conference on Artificial Intelligence, Machine Learning, and Business Analytics, 2021 Marketing Modelers' Meeting, 2021 Marketing Science Conference, and 2021 virtual academic job interviews.

[‡]Associate Professor, Hunan University, School of Public Administration, Lushan South Road, Changsha, China, husaiquan@hnu.edu.cn.

[§]John D. C. Little Professor of Marketing, Massachusetts Institute of Technology, MIT Sloan School of Management, 100 Main Street, Cambridge, MA 02142, jjzhang@mit.edu.

[¶]Ph.D. candidate in Management, Massachusetts Institute of Technology, MIT Sloan School of Management, 100 Main Street, Cambridge, MA 02142, yzhu44@mit.edu.

1 Introduction

Over 13 million Americans worked in sales-related jobs in 2020, representing nearly 10% of the labor force.[1] Without requiring an advanced or even a college degree, the sales occupation is considered accessible to many. “Sales is the hardest easy job in the world,” says Bob Franco, author and veteran of sales (Franco 2015). Yet, sales is the *hardest* easy job. It is by far one of the most stressful occupations according to a PayScale survey (Hedges 2016). Starting a sales career is particularly challenging. For new salespeople, the process of converting strangers into paying customers is fraught with failure. In the company we partnered with for this study (discussed in more detail in subsequent sections), 70% of new salespeople failed to make a sale 90 days into their job.

New salespeople’s productivity hurdle can be costly to companies. Sales is already the most expensive marketing function. North American companies spend $1.2 trillion on sales each year (Ahearne 2019).[2] U.S. companies spend $15 billion a year on sales force training alone and $800 billion on sales force incentives (Sunder et al. 2017). New salespeople’s productivity hurdle undermines these hefty investments. It affects company revenue and customer experience. Moreover, it threatens sales force retention. By February 2021, 91% of the salespeople who had joined our partner company since 2015 had left. Notably, 67% left without ever making a sale, citing the frustration of getting started as a common reason to quit.[3]

In this paper, we develop a deep learning based recommender system to help new salespeople improve their productivity. We focus on “sales prospecting.” For each new salesperson, we offer personalized recommendations of customer types (e.g., female customers below 40) with higher conversion potential, based on deep learning of historical data on sales revenue, salesperson traits, and customer traits.

[1] Source: U.S. Bureau of Labor Statistics (https://www.bls.gov).

[2] This is compared with $165 billion annual spending on traditional marketing and $36 billion on digital.

[3] For context, the annual sales force turnover rate is twice the average of the entire labor force (source: https://hbr.org/2017/07/how-to-predict-turnover-on-your-sales-team).

This approach suits the sales context for the following reasons. First, new salespeople often need help recognizing promising customers. In a recent HubSpot report, over 40% of salespeople find sales prospecting the most difficult part of their job.[4] Second, sales outcome often depends on the match between the salesperson and the customer on various traits such as communication style (Williams and Spiro 1985), gender (Lieven 2016), and race (Richard et al. 2017). Recommender systems are particularly effective at uncovering such personalized match value between users and recommended items (Aggarwal 2016). Third, salesperson-customer match can be complex. It may be driven by convoluted interactions of salesperson/customer traits and these traits may be high-dimensional. For example, appearance has been shown to influence trust (e.g., Duarte et al. 2012) whereas different customers may value trust differently. We thus focus on deep learning based recommender systems (e.g., Zhang et al. 2019) to capture the intricacies of salesperson-customer match and to embrace unstructured data (appearance data in our application).

What is new in our approach is that we extend standard deep learning based recommender systems to capture an important feature of the sales context – salespeople's failures, which are prevalent but often do not show up in historical sales records. For instance, it may appear that a salesperson has never made a sale to female customers below 40. In standard deep learning based recommender systems, these instances are often treated as missing observations and excluded from model training (e.g., Lu et al. 2015, Ricci et al. 2015, Aggarwal 2016, Chen et al. 2021). We emphasize a different method. We argue that these instances are substantively meaningful; they not only should be included in training, but also included in a theoretically constructed way to capture the behavioral process they represent.

More specifically, we emphasize that sales records may be *missing by choice* for some salesperson-customer dyads. It could be that a salesperson tried but failed to convert

[4]Source: https://blog.hubspot.com/sales/sales-statistics. Salespeople are responsible for customer prospecting in our partner company and many others. However, as we will discuss, our approach extends to contexts where companies assign prospects to salespeople.

a customer type. It could also be that a salesperson decided not to sell to a customer type because the cost of selling exceeded the expected gain. Either way, the fact that the salesperson did not sell to a customer type is informative; a new and similar salesperson should probably avoid this customer type. Missing-by-choice sales records can be particularly meaningful in the sales function. They allow new salespeople to learn from not only their predecessors' success but also their lack of success.

Computationally, we develop an *augmented recommender system* to explicitly account for missing-by-choice sales records in deep learning based recommender systems. We specify a parsimonious behavioral model in which a salesperson sells to a customer type only if the expected gain outweighs the cost. We incorporate this behavioral model into a neural network structure by proposing a new activation function, which outputs a sales record only if sales revenue exceeds a cost threshold to be estimated. This allows us to augment standard deep learning based recommendation systems in a simple, interpretable, and flexible way.

We train, validate, and test our augmented recommender system using data from a major insurance company headquartered in Shanghai. The data include 12,149 salespeople and 409,840 insurance transactions from January 2015 through February 2021. Our augmented recommender system significantly outperforms a set of benchmarks on two common criteria of recommender system performance: prediction accuracy (mean square error) and recommendation quality (F1 and NDCG scores, to be explained later). In particular, our method significantly outperforms deep learning based recommender systems that either exclude missing sales records from model training or retain them but use the standard linear activation function.

Because our recommender system augmentation is based on modeling the sales-data generating process, it can be easily extended to accommodate a range of scenarios depending on why sales records are missing. We present two such extensions, where salespeople may not get to consider all customer types and where the cost threshold may be heterogeneous across salespeople. Both extensions further improve our method's recommendation quality.

Finally, simulation results suggest that our augmented recommender system can be practically valuable. Compared with a naïve, random search strategy that new salespeople may follow, our recommender system may reduce the number of failures before the first sale by as high as 40%. In doing so, our recommender system can reduce the proportion of unproductive new salespeople by 27% 90 days into their job.

Our contributions are two-fold. Substantively, we develop a data-driven process to help new salespeople recognize suitable prospects and help companies improve sales force management. In doing so, we showcase a novel application of recommender systems in the sales context. Methodologically, we augment deep learning based recommender systems by retaining, modeling, and retrieving information from missing-by-choice observations. We demonstrate the efficacy of the augmented recommender system for sales prospecting, but we expect it to be broadly relevant wherever historical records for some user-item dyads are missing by choice, as opposed to missing by chance. For example, if a Netflix user has never rated a movie genre or if an Amazon customer has never bought from a product category, our augmented recommender system can be applied to derive information from these seemingly missing records. In the following section, we discuss our contribution to the literature in more detail.

2 Related Literature

Sales force management is central to marketing research. Sales force compensation design in particular has attracted significant research and generated extensive insights on how to motivate salespeople (see Misra 2019 and Chung et al. 2020 for comprehensive reviews). To the extent that compensation policies are not always feasible to change (Daljord et al. 2016), another line of papers explore non-monetary sales force management strategies. These include optimizing the composition of sales teams (Chen and Lim 2017), empowering salespeople with genetic self-knowledge (Gong et al. 2021), and training sales agents using artificial-intelligence coaches (Luo et al. 2021). Our paper contributes another non-monetary sales force management tool, a recommender system that helps salespeople recognize suitable prospects.

Recommender systems are a powerful way to help individuals choose among many options, often drawing on their many peers' choices (Ricci et al. 2015, Aggarwal 2016). Ansari et al. (2000)'s seminal paper introduces recommender systems to marketing and develops a Bayesian preference framework for recommendation. A stream of marketing papers have extended and applied recommender systems along important dimensions, such as optimizing purchase lift (Bodapati 2008), personalizing content offerings adaptively (Chung et al. 2009), generalizing consumer preference models for experienced goods (Chung and Rao 2012), eliciting consumer preferences for complex products (Huang and Luo 2016), modeling topics to leverage rich product information (Ansari et al. 2018), recommending options to help consumers learn their preferences (Dzyabura and Hauser 2019), using consumer search data (Gardete and Santos 2020) or on-boarding surveys (Dew 2021) to overcome the cold-start problem, and evaluating welfare implications of personalized rankings (Donnelly et al. 2022).[5]

We contribute to the recommender-system literature by developing an augmented, deep learning based recommender system to incorporate missing-by-choice data. A particularly relevant paper in marketing is Ying et al. (2006), who address the selection bias in consumer ratings with a hierarchical Bayes model that jointly captures rating value and rating incidence. In computer science, although most recommender-system papers focus on ways to better fit data, there is a line of research on debiasing data (see Chen

[5]The cold-start problem refers to the lack of historical data on user-item interactions to inform recommendation. In our paper, new salespeople are cold-start users by definition. We overcome this challenge using deep learning based recommendation, drawing on "side information" about salespeople (e.g., demographics, appearance). In a related paper, Padilla and Ascarza (2021) address the cold-start problem in customer relationship management, using probabilistic machine learning to incorporate customer side information.

et al. 2021 for a systematic review). Solutions include probabilistic models (Hernandez-Lobato et al. 2014) and propensity scores (Schnabel et al. 2016), both in the context of matrix factorization based recommender systems.[6] Our paper shares the same view that there is information to be gained from including missing data and modeling why they are missing. The difference is that we develop a way to incorporate missing-by-choice data in deep learning based recommender systems.

Deep learning based recommender systems are becoming prevalent in both academic research and industry use (see Zhang et al. 2019 for a review). Successful applications include the recommender systems for YouTube videos (Covington et al. 2016) and Google Play apps (Cheng et al. 2016), among many others. Deep learning allows recommender systems to capture the potentially complex, nonlinear relationship between users and items and to handle various forms of unstructured data (Goodfellow et al. 2016). Deep learning can also be integrated with other methodologies, fueling innovations such as neural network matrix factorization (e.g., Dziugaite and Roy 2015) and Bayesian deep learning (e.g., Wang and Yeung 2020). A major downside of deep learning based recommendation though is its lack of explainability; its numerous parameters and activation functions are often not easily interpretable (Rajaram and Manchanda 2020, Zhang and Chen 2020). In addition, overparameterized deep learning models may overfit the training data and fail to generalize out-of-sample (Zhang et al. 2021).

Our augmentation can thus be valuable in three ways. First, as we show, this augmentation can further improve the performance of deep learning based recommender systems in situations where data are missing by choice – and these situations are likely the norm rather than the exception (Marlin et al. 2007). Second, our augmentation is interpretable; at its core is an activation function based on a microfounded user behav-

[6]Matrix factorization is a technique to decompose users' reactions to various items into lower-rank user and item matrices. Matrix factorization based recommender systems are popular in industry (e.g., Koren et al. 2009, Gower 2014, Gomez-Uribe and Hunt 2015). However, they are often subject to challenges such as the cold-start problem (e.g., Bobadilla et al. 2012, Wei et al. 2017), ability to handle complex unstructured data (e.g., Chu and Tsai 2017, Wang et al. 2017), and scalability (e.g., Najafabadi et al. 2015).

ior model.[7] Because of this feature, as we show, the augmentation can be extended in interpretable ways to capture different user behaviors. Third, our augmentation can be easily implemented. In its simplest case, it requires only one additional parameter to be estimated (the cost threshold). This helps maintain the scalability of the recommender system and mitigate overfitting concerns.

Last, our paper is broadly related to the literature on using machine learning for personalization and targeting. Experimental data are often used for strategy evaluation in this literature (Dube and Misra 2019, Simester et al. 2020, Gabel and Timoshenko 2021). Although experiments offer a clean way to assess causal treatment effects, they may constrain the number of personalization or targeting actions that can be evaluated in one study. By contrast, we use historical sales data for model training and evaluation. There is no theoretical upper limit to the number of recommendations that can be tested. This approach is in line with Hauser et al. (2009) and Yoganarasimhan (2020), who use clickstream data to study large sets of personalized search actions. As such, the focus of our paper is prediction, consistent with the literature summarized in Proserpio et al. (2020), as opposed to treatment effect estimation. However, these two approaches can complement each other. For instance, predictive models can help select a feasible number of high-potential recommendation strategies to be tested experimentally.

3 An Augmented Recommender System

In this section, we present the construction of our augmented recommender system. As an overview, we incorporate a model of missing-by-choice sales records into a neural network framework. We first present the model and then the neural network adjustment.

[7] In a recent study of brand selfies, Hartmann et al. (2021) also modify and add layers to a neural network. Their goal is better classification and interpretation.

3.1 Model of Missing-by-Choice Sales Records

We strive to specify a parsimonious model to capture the data generation process behind sales records, including the process by which a subset of them are missing. We distinguish between two types of sales records: latent sales and observed sales. Latent sales represent the underlying sales outcome for each salesperson-customer dyad. Observed sales are the sales records that actually appear in the data. Our model aims to describe the latent sales generation process and the relationship between latent and observed sales.

Let $y^*_{ij} \in \mathbb{R}$ denote the latent sales revenue generated by salesperson $i \in \{1, ..., I\}$ from selling to customer type $j \in \{1, ..., J\}$.[8] Both I and J are finite numbers but can be large. Suppose u_i represents traits of salesperson i and v_j represents traits of customer type j. Both u_i and v_j can include numerical features (e.g., age, income), categorical features (e.g., gender, education, occupation), and features extracted from unstructured data (e.g., facial images). The latent sales generation process follows:

$$y^*_{ij} = f(u_i, v_j; \theta). \tag{1}$$

The $f(\cdot|\theta)$ function maps salesperson i's traits u_i and customer type j's traits v_j to the latent sales y^*_{ij} associated with the salesperson-customer dyad ij. $f(\cdot|\theta)$ is parameterized by θ and can take any functional form. As such, $f(\cdot|\theta)$ can flexibly capture the effect of salesperson-customer match, which has been shown to influence sales outcome.

Next, let y_{ij} denote the observed sales revenue salesperson i generated from selling to customer type j. We specify the relationship between latent and observed sales as:

$$y_{ij} = \begin{cases} y^*_{ij} & \text{if } y^*_{ij} > c \\ 0 & \text{otherwise} \end{cases}. \tag{2}$$

[8]Customer type is a generic reference in our model. It can be as granular as a specific customer, or coarsened to describe a group of customers who share certain traits. We will discuss the operationalization of customer type in subsequent sections. We call the combination of a salesperson and a customer type a salesperson-customer dyad for brevity.

In other words, we posit that if the latent sales y^*_{ij} exceeds a cutoff c, it becomes an observed sales record. Otherwise, the observed sales record equals zero, meaning that salesperson i has never made a sale to customer type j.

To interpret this data generation process, imagine a salesperson who decides whether to make an effort to sell to a customer type. The cost of selling includes the time cost, effort cost, and opportunity cost. The benefit of making a sale includes commissions from the latent sales revenue (which in our setting is prespecified and known to the salesperson) and any psychological payoff (Thaler 2008). This benefit is further scaled by the perceived probability of sale and adjusted for risk preferences (Rabin 2000), time preferences (Loewenstein et al. 2003), and prediction errors (Mullins et al. 2014). As such, the key parameter $c \in \mathbb{R}$ represents the adjusted, scaled, net cost of selling, referred to as "net cost" hereafter for brevity. The salesperson will choose not to sell to the customer type if the benefit is below cost or, equivalently, if the latent sales revenue y^*_{ij} is below net cost c. This creates a missing-by-choice sales record for the salesperson-customer dyad ij.

Analogous arguments hold if a salesperson tries to convert a customer type based on prior cost-benefit analysis but fails because of higher-than-expected selling cost or worse-than-expected conversion probability. The same intuition applies – the lack of sales records for this salesperson-customer dyad likely implies that the associated latent sales revenue is below (realized) net cost.

By modeling missing-by-choice sales records, we uncover and preserve the information value of these seemingly missing observations. To the extent that the match of salesperson-customer traits matters, salesperson i's lack of sales records with customer type j means that, other things being equal, new salespeople who are similar to salesperson i should expect lower latent sales revenue from serving customer type j. The augmented recommender system will recognize this and will be less likely to recommend customer type j to these new salespeople.

We have three comments. First, our model resembles the classic tobit model of truncated observations (Tobin 1958).[9] Our model also echos the well-known discrete-continuous models in marketing (e.g., Chintagunta 1993, Ying et al. 2006) in that discrete events (e.g., purchase incidence, decision to leave a rating, failure to serve a customer) and continuous quantities (e.g., purchase volume, rating value, sales revenue) can jointly reveal the same underlying decision process. We are humbled to build on these established theories to improve the design of neural network structures for better performance.

Second, we do not explicitly model sales effort. However, under the common assumption that salespeople in equilibrium optimize their effort choices given their own and customers' traits, the mapping function $f(\cdot|\theta)$, which has no functional form restrictions, can be seen as already subsuming the effect of effort (Laffont and Martimort 2009, Misra and Nair 2011, Chung et al. 2014).

Third, we intentionally assume a homogeneous net cost c across salespeople and customer types for the main model. This allows us to keep the model as simple as possible to isolate the mere effect of considering missing-by-choice sales records, as opposed to the effect of adding more parameters. We extend the analysis to accommodate heterogeneous net costs across salespeople in Section 6.2.

3.2 Neural Network Framework

We now incorporate the model of missing-by-choice sales into a neural network framework. We keep our presentation intuitive and refer interested readers to Goodfellow et al. (2016) for an excellent text on the foundation and applications of deep learning.

A neural network is based on a collection of connected units called neurons. A neuron can process "signals" passed into it and then pass the output signals into its connected neurons. A signal is a real number. An activation function computes the output of a neuron based on its inputs. The connections are called edges. Neurons and edges have

[9] The difference is that the tobit model captures mass points in otherwise-continuous dependent variables, whereas our observed sales function is allowed to be discontinuous at the mass point of zero sales.

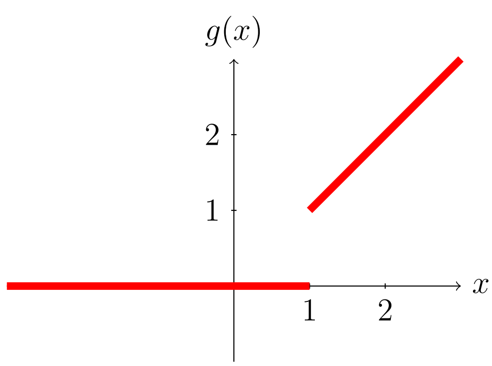
weights that adjust as learning proceeds. Typically, neurons are aggregated into layers. Figure A1 of the Online Appendix illustrates a typical neural network structure where data pass from the input layer to the output layer, possibly after travelling through a series of middle hidden layers.

We propose a new activation function to be used in the output layer. The idea is to utilize the hidden layers of the neural network to flexibly capture the latent sales generation process of Equation (1) and then adjust the output sales by imitating the observed sales generation process of Equation (2). The new activation function is:

$$g(x) = \begin{cases} x & \text{if } x > c \\ 0 & \text{otherwise} \end{cases} . \tag{3}$$

Recall that c is the net cost that governs whether latent sales records are observed. Computationally, c is a hyperparameter to be tuned during neural network optimization (Lau and Lim 2018). Note that when c equals zero, our proposed activation function is the same as the well-known Rectified Linear Unit (ReLU) activation function: $g(x) = \max\{0, x\}$.[10] In Figure 1, we offer an illustration of our activation function when c differs from zero.

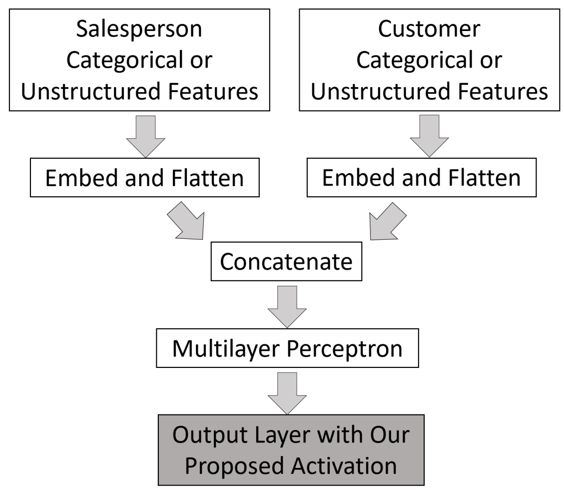
Figure 1. Proposed Activation Function (Example of $c = 1$).



[10]When $c \neq 0$, our activation function is different from the ReLU activation function where the input is renormalized to $\hat{x} = x - c$. ReLU activation is a continuous activation function, whereas our proposed activation function is discontinuous when $c \neq 0$. The error back-propagation learning algorithm is suitable when the activation function is discontinuous (Findlay 1989).

Figure 2 summarizes the complete neural network structure underlying our augmented recommender system. The main augmentation is that we use our proposed activation function in the output layer. Other layers' structures remain standard for a regression problem using neural network models (Athey and Imbens 2019).

Figure 2. Our Proposed Neural Network Structure.



Specifically, we first embed each categorical or unstructured feature of either salespeople or customer types and then flatten the embedding results. Embedding is a method used to represent data as a vector of continuous numbers, whereby the continuous vector can meaningfully represent the data in the transformed space (Cui et al. 2019). The unstructured data we use in this paper are salespeople's facial image data. Specifically, a face embedding is a vector of continuous numbers that represents features extracted from the facial image. We then concatenate all the embedding vectors and pass the concatenating results into a multilayer perceptron (MLP). An MLP is a class of feedforward neural networks (ANNs) and a universal function approximator per Cybenko's theorem (Cybenko 1989). The goal of the MLP in our framework is to capture the complex function $f(\cdot|\theta)$ that maps salesperson and customer traits into latent sales. The MLP we consider has two hidden layers with ReLU activation at each layer. We also use Dropout at each layer to prevent neural networks from overfitting (Srivastava et al. 2014). Last, we pass the results from the MLP to the final output layer with our proposed activation function as specified in Equation (3). More details on the neural network structure can be found in Online Appendix A.2.

Three comments are in order. First, the neural network structures before the output layer are not fixed and can be adapted easily based on the type of input data. For example, if voice data of salespeople and/or customers become available, long short-term memory recurrent neural networks (LSTM RNNs, Graves et al. 2013) can be integrated with our proposed activation function in the output layer.

Second, the way we model missing-by-choice sales for the neural network serves as an example of using domain theories to improve deep learning algorithms. At a fundamental level, our approach shares the same essence of the invention of Dropout (Srivastava et al. 2014), Dual Rectified Linear Units (DReLUs, Godin et al. 2018), and Attention in Bidirectional Encoder Representations from Transformers (BERT, Devlin et al. 2019). The difference is that our technical modification is motivated by a microfounded model of how agents behave and how their behaviors shape the observed data.

Third, in our proposed framework, sales records are missing because a salesperson chooses not to sell to a customer type or fails to do so. As previously mentioned, the assumption is that the salesperson has considered all customer types. We address this remaining issue in three ways. First, in our main analysis in Section 5, we develop the recommender system focusing on experienced salespeople, who have arguably considered all customer types. Second, we intentionally classify customers into a reasonable number of types, such that a salesperson can feasibly consider all types. Finally, in extended analysis of Section 6.1, we consider and model the probability that a salesperson has not considered all customer types.

4 Data

In this section, we present the data we will use to develop and evaluate the augmented recommender system. We first introduce the business context. We then describe the data in detail. Last, we present data patterns that suggest that sales records may be missing by choice and that salesperson-customer match may influence sales outcome.

4.1 Business Context

We collaborate with a major insurance company headquartered in Shanghai, China. Similar to its counterpart in the U.S., the sales profession in China is considered to be open to people of various backgrounds, while offering potentially good monetary reward. The average yearly gross salary for salespeople is around 20,000 USD,[11] while the per capita disposable income, as of 2020, is around 5,000 USD in China.[12]

The company focuses on providing life insurance, especially critical-illness insurance products, targeting the low- and mid-tier markets. As of 2021, the company has had hundreds of thousands of employees, established thousands of branches in the country, and served over a million customers (exact numbers withheld given the confidentiality agreement). Salespeople in the company are responsible for prospecting customers themselves, often in face-to-face settings.

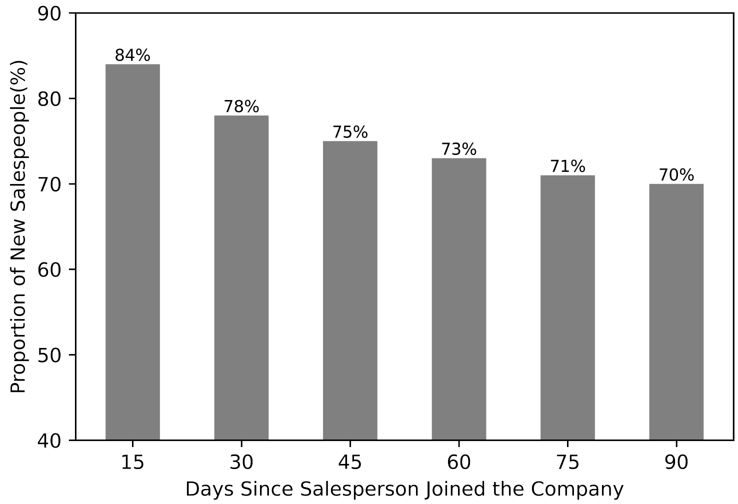
We interviewed the company's management and sales force as background research. The productivity of new salespeople emerged as a major challenge. Figure 3 presents the proportion of new salespeople who have not made their first sales among all 123,836 salespeople who joined the company between January 2015 and November 2020. Three months into their job, 70% of the new salespeople still have not started their first sale. This has a rather negative impact on sales force morale, efficacy, retention, and recruitment.

According to our interviews, the match between salespeople and customers plays an

[11] Source: https://www.erieri.com/salary/job/inside-salesperson/china.

[12] Source: http://www.stats.gov.cn/english/PressRelease/202101/t20210119_1812523.html.

Figure 3. Proportion of New Salespeople Who Have Not Made Their First Sale.



important role in driving sales outcome, whereas new salespeople generally do not know what types of customers are suitable prospects for them. In comparison, how to explain insurance products to customers does not seem to be a challenge. Sales force compensation includes a base salary and a fixed percentage of sales revenue. The compensation structure is practically difficult to change (Daljord et al. 2016 make similar observations). In addition to managerial constraints, the company is supervised by the China Insurance Regulatory Commission and its compensation structure is strictly regulated. Therefore, we focus on non-monetary sales force management strategies and develop a recommender system to help new salespeople recognize suitable prospects.

4.2 Sample, Variables, and Summary Statistics

The company provided all sales records from January 2015 through February 2021, in the form of unique contracts. Each contract documents the associated salesperson's collectable and storable information, which includes age (when the salesperson joined the company), gender, education, home province, branch served, title in the company, years worked at the company, whether the salesperson was referred to join the company, and whether the salesperson had left the company by February 2021.

We also have access to facial image data that salespeople in our sample consented to share and use. The social psychological literature has shown that facial appearance affects perceived trustworthiness (e.g., Chang et al. 2010, Stirrat and Perrett 2010, Sofer et al. 2015).[13] Meanwhile, trust is known to shape sales outcome (e.g., Ganesan and Hess 1997, Urban et al. 2000, Urban et al. 2009). Therefore, We include facial image data as a potentially useful feature in our recommender system.[14]

Besides salesperson information, each contract documents the customer's collectable and storable information, which includes age (at the time of transaction), gender, marital status, occupation, and relationship with the insured. We use these variables to characterize a customer type. Finally, each contract documents the associated sales revenue, calculated as a standard premium of each transaction, following industry norm. This serves as the salesperson's key performance index.

As discussed in Section 3.2, we develop our main recommender system using data of experienced salespeople because they likely have considered all customer types. Thus, we restrict attention to salespeople who joined the company between January 2015 and July 2020, and who have completed at least five sales. These criteria are suggested by top management based on knowledge of what makes an experienced salesperson in the company. This yields 12,149 salespeople and 409,840 insurance sales records to construct and evaluate our recommender system. Repeat purchase is rare for the company. We assume each sale is for a different customer for tractability. On average, each salesperson in the data has achieved 34 sales.

Table 1 presents the summary statistics of structured salesperson and customer traits

[13]This effect can be consequential in a broad range of domains, such as lending (Duarte et al. 2012), CEO selection (Stoker et al. 2016), science communication (Gheorghiu et al. 2017), career development (Malik et al. 2020), and targeting effectiveness (Tkachenko and Jedidi 2020).

[14]In doing so, we must eliminate salespeople without image data from our model's training, validation, and testing. These salespeople account for about 1/3 of the entire sample. We acknowledge potential self-selection into image-data provision and caution against using our recommender system without adjustment for salespeople who do not provide image data.

in the data.[15] There are several observations to note. The average age of salespeople when joining the company is 38. This is likely an age with much family responsibilities, making job success particularly helpful. Meanwhile, most of the salespeople have not attended college. This echos the general observation of the sales profession being accessible. In addition, the most frequent customer occupation is farmer, consistent with the company's focus on serving low- to mid-tier markets.

Table 1. Summary Statistics of Salesperson and Customer Traits.

Variable	Mean	# Types	Most Frequent Type
Salesperson Traits (N = 12,149)			
Age (When Joining Company)	38	45	37
Gender (Female = 1)	0.74	2	1
Education	-	7	High School
Home Province	-	33	Hebei
Branch Served	-	25	Hebei
Title in Company	-	7	Entry Level
Years Worked at Company	1.4	7	1
Whether Referred to Join (Referred = 1)	0.98	2	1
Whether Left Company (Left = 1)	0.1	2	0
Customer Traits (N = 409,840)			
Age (at Time of Transaction)	38	64	32
Gender (Female = 1)	0.58	2	1
Marital Status	-	4	Married
Occupation	-	1100	Farmer
Relationship with the Insured	-	26	Self

As Table 1 suggests, using raw data on customer traits yields an enormous number of customer types. Including too many customer types in the recommender system can be problematic. First, as discussed in Section 3.2, it is implausible that a salesperson, even an experienced one, has considered this many customer types. Second, it is impractical to offer highly granular recommendations (e.g., a 45-year-old female lawyer) to new salespeo-

[15] For the unstructured data in our sample (salesperson facial image data), one approach is to derive handcrafted features (e.g., facial width) and then input them into the recommender system. We take a different approach; we input the raw image data to exploit the joint end-to-end representation learning advantage of deep learning based recommender systems (Zhang et al. 2019). Therefore, we do not present summary statistics of the image data. Following confidentiality agreement, we will not report summary statistics of sales revenue either.

ple. Therefore, we dichotomize customer age as weakly above or below 40, marital status as married or not, and relationship with the insured as self-insured or not. Finally, we follow the Chinese Occupation Classification and group customer occupations to seven types.[16] This yields 112 customer types in total. Figure A7 in the Online Appendix reports the distribution of customer types for each customer trait. The recommender system will output one or multiple, depending on system design, of these 112 customer types for each new salesperson. An example would be: a female, married customer, who is above 40, works as a technical staff, and is interested in insuring herself.

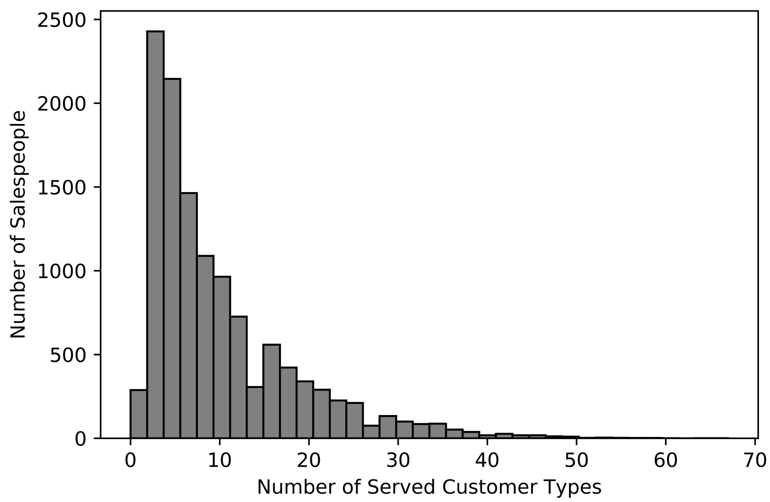
4.3 Preliminary Data Patterns

We examine the data for distributional insight across customer types. Figure 4 presents a histogram of the number of customer types each salesperson in our data has sold to. Most of the salespeople served a small proportion (less than 15%) of the 112 potential customer types. In the most common case, a salesperson served less than ten customer types. Recall that these are experienced salespeople who arguably have considered all customer types. As such, there are possibly certain customer types that certain salespeople either chose not to serve or failed to serve. These would lead to missing-by-choice sales records in our data.

We also look at the customer type that brings the highest sales revenue for each salesperson in the data. Across the 12,149 salespeople, there is hardly one customer type that brings the most sales to all salespeople; rather, these "most valuable" customer types are dispersed, spanning 106 out of the 112 customer types in the data. Furthermore, for each of the 112 customer types in the data, we calculate the percentage of salespeople who

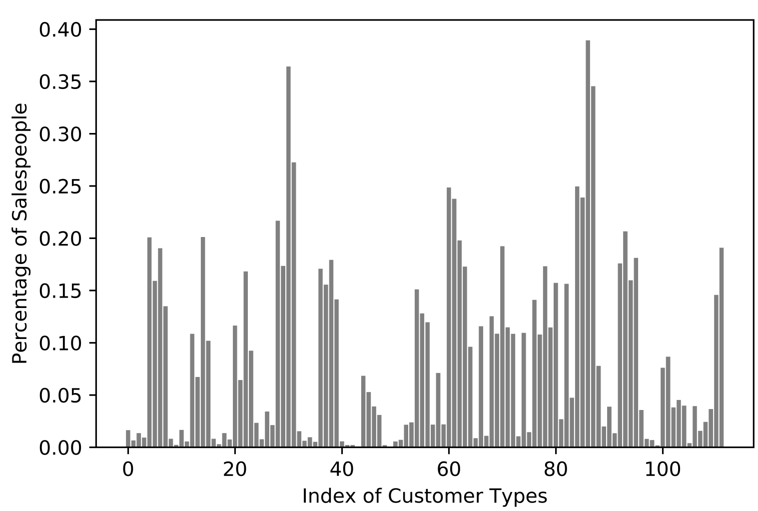
[16] The seven occupation types are: 1) managers who work in government, business, and nonprofit organizations; 2) technical staff (e.g., doctors, lawyers, actors/actresses, researchers); 3) office staff and public service providers (e.g., administrative staff, administrative arbitrators, policemen, firemen); 4) service personnel (e.g., waiters/waitresses, salespeople, babysitters, doormen); 5) farmers, including personnel working in agriculture, forestry, animal husbandry, and fishery; 6) production workers in various sectors (e.g., furniture manufacturing, textile, coal chemical production, drug manufacturing); 7) soldiers and others.

Figure 4. Most Salespeople Served Just A Few Customer Types.



have made sales to this customer type. As Figure 5 shows, none of the customer types can attract all salespeople. There is also noticeable heterogeneity in the percentage of salespeople each customer type tends to attract. These results suggest that there is no customer type that suits all salespeople. Therefore, a personalized recommender system may add value, compared to a simpler recommendation to target a specific customer type.

Figure 5. Percentage of Salespeople Who Served Each Customer Type.



5 Evaluating the Augmented Recommender System

We evaluate our augmented recommender system in this section. In particular, we test whether incorporating missing-by-choice sales records improves deep learning based recommender system performance. We focus on two commonly used performance criteria for recommender systems: prediction accuracy and recommendation quality. In addition, we present a set of interpretable outputs of our recommender system.

5.1 Prediction Accuracy

For prediction accuracy, we use the standard metric of mean squared error (MSE):

$$\text{MSE} = \frac{1}{IJ}\sum_{i=1}^{I}\sum_{j=1}^{J}(y_{ij} - \hat{y}_{ij})^2. \tag{4}$$

The elements of this equation are defined in Section 3. Each data point represents an interaction between a salesperson (indexed by i) and a customer type (indexed by j), including dyads where the salesperson did not sell to the customer type.

Two comments are in order. First, we operationalize y_{ij} using the average daily sales revenue associated with salesperson i and customer type j during the time span of our data. Salespeople vary in how often they sell to a customer type and how much revenue they derive from each sale. Taking the average of sales revenue over time captures the overall value of each customer type to each salesperson that encompasses factors such as the density of this customer type in the population, the ease of conversion, and the gain from a conversion.

Second, we focus on observed sales (including zero), as opposed to latent sales, as the outcome measure. The observed sales revenue y_{ij} comes from the historical transaction data by definition. The predicted sales revenue $\hat{y}_{ij}$ is generated by the recommender system. In our augmented recommender system, $\hat{y}_{ij}$ predicts what sales revenue salesperson

i expects to derive from customer type j, allowing for the possibility that salesperson i does not sell to customer type j. For a new salesperson, our system will recommend the customer type(s) that will yield the highest predicted sales revenue. This allows the recommendation to convey the overall potential of each customer type, after considering the net cost of selling.

We have a total of 1,360,688 salesperson-customer dyads (data points) for the $I = 12,149$ salespeople and $J = 112$ customer types. We randomly split these data into training data (60%), validation data (20%), and test data (20%). We use the training data to train the recommender system, the validation data for cross-validation tuning, and the hold-out test data to evaluate the recommender system's performance. Out of these salesperson-customer dyads, 1,239,345, or 91%, have no sales records.

We consider two benchmark recommender systems. For fair comparison, both benchmarks are deep learning based recommender systems built on the same neural network structure as our augmented recommender system. The first benchmark, which we call "Deep Learning — Missing Excluded," uses only observed sales records for training. This benchmark serves to test the information value of missing-by-choice sales. We scale the sample size of the benchmark to be the same as our method via random resampling. This ensures that our method's performance improvement, if any, occurs not simply because it includes more data points.

The second benchmark, "Deep Learning — Linear Activation," uses the same training data as our method. All the neural network components before the output layer are also the same. The only difference is that, in the output layer, this benchmark uses the standard linear activation function, as opposed to the activation function we propose in Equation (3). This benchmark serves to test whether our new activation function has value in capturing missing-by-choice sales records.[17]

[17] We do not explicitly examine the benchmark of a deep learning based recommender system using the ReLU activation. This is because, as discussed, our method nests the ReLU activation as a special case where the net cost c is zero. Whether c is zero is an empirical question, whereas our method offers c, being it zero or not, a behavioral interpretation.

Table 2 presents the MSE on the test data for the two benchmarks and for our proposed augmented recommender system, referred to as "Deep Learning — Augmented." We normalize the MSE of the augmented recommender system to be 100, following our confidentiality agreement with the company. For this and subsequent metrics, we use block bootstrapping (Horowitz 2001) to derive standard errors and test the significance of differences between a benchmark and our augmented recommender system.

Table 2. Prediction Accuracy.

Recommender System	MSE	
	(1)	(2)
	All Test Data	Positive Sales
Deep Learning — Missing Excluded	139.84***	101.92***
Deep Learning — Linear Activation	100.91***	100.56
Deep Learning — Augmented (Our Method)	100	100
# Observations	272,138	24,169

Notes. Each observation is a salesperson-customer dyad in the test data. Significance pertains to comparison with our method. *** $p < 0.01$.

Column (1) of Table 2 presents the MSEs using all the test data. Our method shows much greater prediction accuracy, in both significance and magnitude, than the benchmark that excludes missing sales records. This suggests that missing-by-choice sales are highly informative and should be included in recommender system training. Linear activation improves prediction accuracy over the first benchmark, but our method remains significantly more accurate at the $p < 0.01$ level. This shows that our proposed activation function itself has value in increasing prediction accuracy.

Key to our augmentation is our treatment of missing-by-choice sales records. In the spirit of a falsification test, we repeat our MSE calculation only on salesperson-customer dyads in the test data that have positive sales records. This subsample mimics an environment in which each salesperson gets to serve all customer types. Column (2) of Table 2 shows the results. In this case, the three methods have similar magnitudes of MSEs. Moreover, the difference between our method and the linear activation benchmark is statistically insignificant with $p = 0.31$. Therefore, as expected, our augmentation is more valuable in markets where missing-by-choice sales are common.

5.2 Recommendation Quality

We next evaluate the recommendation quality of our augmented recommender system. We consider two common evaluation metrics in the literature (e.g., Steck 2013, Silverira et al. 2019, Yoganarasimhan 2020): F1-score and Normalized Discounted Cumulative Gain (NDCG). Intuitively speaking, the F1-score measures whether a recommender system is good at identifying items (e.g., customer types), whereas NDCG measures whether a recommender system is good at identifying and ranking items.

The F1-score depends on both "precision" and "recall." Precision is defined as the number of "relevant" recommendations divided by the total number of recommendations. In our setting, we define a recommendation as relevant if the salesperson (in the test data) has indeed had a transaction with the recommended customer type. Recall is defined as the the number of relevant recommendations divided by the total number of relevant customer types. The F1-score balances precision and recall to achieve a single-dimension comparison between different recommender systems:

$$\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

NDCG further takes into account the ranking among recommended items. The idea is that highly relevant customer types should ideally be highly ranked in the list of recommended types. We quantify the relevance score by using the actual daily sales revenue for each salesperson on each customer type. To calculate NDCG, we divide Discounted Cumulative Gain (DCG) by Ideal Discounted Cumulative Gain (IDCG). For each salesperson, the DCG measure can be calculated from the formula, $\text{DCG} = \sum_r \text{Relevance}_r / \log_2(r+1)$, where Relevance_r is the relevance score between the salesperson and the customer type recommended in ranking position r of the list, and the logarithmic function is used to scale the importance of ranking. By arranging customer types in descending order of actual daily sales revenue, we can derive IDCG using the same formula as DCG.

We again compare our method with the two benchmark recommender systems: Deep Learning — Missing Excluded and Deep Learning — Linear Activation. In addition, we consider a non-personalized recommender system that simply recommends the top customer types based on their associated historical sales revenue. Both the F1-score and NDCG depend on the number of recommendations a system is designed to offer. We vary this number from one to three. In addition, both the F1-score and NDCG are salesperson-specific. We first compute their values for each salesperson in the test data and then take the average across these salespeople.

Table 3 presents the F1-score and NDCG results. For both metrics and for all numbers of recommendations considered, excluding missing sales records from training leads to the worst recommendation quality. Linear activation improves recommendation quality and our method achieves further, significant improvement ($p < 0.01$). Our method also significantly outperforms the non-personalized recommender system. This again shows that salesperson-customer match matters for sales outcome.

The goal of our recommender system is to leverage experienced salespeople's sales records (and the lack of them) to help new salespeople. So far, we have evaluated our average recommendation quality for all salespeople in the test data. It remains to check our recommendation quality for new salespeople. In Table 4, we present the F1-score and NDCG for salespeople with different numbers of years working at the company, fixing the number of recommendations at one. Table A1 in the Online Appendix shows the results with two and three recommendations, respectively. Reassuringly, our method continues to report significantly higher recommendation quality than the benchmarks across the board, including the case of relatively inexperienced sales people.

Table 3. Recommendation Quality.

Recommender System	# Recommendations		
	1	2	3
		F1-Score	
Non-Personalized Recommender System	0.044***	0.075***	0.105***
Deep Learning — Missing Excluded	0.025***	0.044***	0.058***
Deep Learning — Linear Activation	0.074***	0.122***	0.155***
Deep Learning — Augmented (Our Method)	0.081	0.132	0.165
		NDCG	
Non-Personalized Recommender System	0.241***	0.305***	0.351***
Deep Learning — Missing Excluded	0.140***	0.187***	0.215***
Deep Learning — Linear Activation	0.371***	0.452***	0.481***
Deep Learning — Augmented (Our Method)	0.385	0.462	0.490
# Observations	2,430	2,430	2,430

Notes. Each observation is a salesperson in the test data. Significance pertains to comparison with our method. *** $p < 0.01$.

Table 4. Recommendation Quality by Salesperson Experience (One Customer Type Recommended).

Recommender System	Years Worked at Company		
	0	1	2-6
		F1-Score	
Non-Personalized Recommender System	0.046***	0.046***	0.041***
Deep Learning — Missing Excluded	0.023***	0.025***	0.027***
Deep Learning — Linear Activation	0.081***	0.074***	0.070***
Deep Learning — Augmented (Our Method)	0.085	0.085	0.074
		NDCG	
Non-Personalized Recommender System	0.126***	0.209***	0.361***
Deep Learning — Missing Excluded	0.065***	0.107***	0.231***
Deep Learning — Linear Activation	0.220***	0.298***	0.560***
Deep Learning — Augmented (Our Method)	0.231	0.316	0.572
# Observations	671	884	875

Notes. Each observation is a salesperson in the test data. Significance pertains to comparison with our method. *** $p < 0.01$.

5.3 Recommender System Output Interpretation

Although we focus on prediction accuracy and recommendation quality, we report the results of our recommender system in this section to facilitate interpretation.

A natural question is which salesperson and customer traits are important. We use the permutation feature importance technique (e.g., Altmann et al. 2010) to answer the question. This technique is especially useful when the model is "opaque," which is often the case for neural network models. Specifically, we take a fitted model and randomly shuffle each variable. We then look at the decrease in the performance measure being considered. Shuffling variables of higher importance causes greater decreases in the performance measure. In Table 5, we present the decrease in MSE (standardized), F1-score, and NDCG, where we implement 100 times of permutation for each variable.

Table 5. Importance of Salesperson and Customer Traits.

Variable	Performance Measure		
	MSE	F1	NDCG
Salesperson Traits			
Facial Image	0.79	0.011	0.022
Age (When Joining Company)	1.33	0.023	0.078
Gender (Female = 1)	0.18	0.005	0.014
Education	0.34	0.003	0.011
Home Province	1.38	0.024	0.090
Branch Served	2.02	0.012	0.034
Title in Company	0.40	0.027	0.101
Years Worked at Company	1.09	0.005	0.010
Whether Referred to Join (Referred = 1)	0.00	0.000	0.000
Whether Left Company (Left = 1)	2.69	0.022	0.074
Customer Traits			
Age (at Time of Transaction)	1.94	0.023	0.067
Gender (Female = 1)	0.14	0.016	0.050
Marital Status	0.80	0.042	0.155
Occupation	3.97	0.054	0.194
Relationship with the Insured	1.17	0.010	0.040

Several results are worth noting. First, whether a trait is important depends on the performance measure. For example, "Title in Company" is the most important salesperson trait when we consider recommendation quality and is not when we consider prediction accuracy. This result highlights the importance of setting the right performance metric for the recommender system. Second, the facial image data do have noticeable impact on recommender system performance across all measures, which underscores the value of using a deep learning based recommender system. Third, across all measures, salespeople's age, home province, branch served, and whether they left the company are important traits, while their gender, education, and whether they were referred to join the company are less important. Last, customer age, marital status, and occupation particularly matter for sales outcome. These results shed light on ways to further improve the recommender system in future research. For instance, it may be worthwhile to categorize customers into finer types along more important traits.

For a recommender system, the ultimate question is what customer type is recommended. The answer depends on the specific salesperson. Note that the results above show the relative importance of each trait in our deep learning model, which subsumes the complex interactions among all traits.[18] To visualize the importance of salesperson-customer match, we present the customer type our model recommends for each salesperson-trait category (e.g., age below 40). Table A2 of the Online Appendix displays the results. To simplify presentation, we examine salesperson traits in isolation, two categories per trait, without including their interactions. For facial image, we split the salespeople into two groups based on clustering of their facial embedding vector values. Even with this coarse classification of salespeople, we can see that match matters. Out of the 10 salesperson traits, only one (i.e., whether referred to join the company) does not affect our model's customer type recommendation. These results again highlight the value of a personalized recommender system.

Finally, cross-validation yields a positive net cost of selling hyperparameter, c. Its magnitude is 4.46% of the average daily sales revenue across all salesperson-customer dyads (value withheld for confidentiality agreements). However, sales revenue is widely dispersed, with a standard deviation 21.78 times its mean value. Therefore, it is possible that a non-negligible fraction of sales revenue falls below the net cost of selling, causing the associated sales records to be missing by choice.

[18] The importance of each trait is analogous to the partial derivative of the outcome measure with respect to this trait, which subsumes any interaction effect between this trait and others.

6 Extensions

Because our augmented recommender system is based on a simple behavioral model, it can be easily extended to capture different market scenarios. In this section, we model two such extensions and discuss a few others.

6.1 Probably Missing-by-Choice Sales Records

For the main analysis so far, we focus on experienced salespeople, assuming that they have had enough time to consider all customer types. In reality, it is possible that even experienced salespeople have ignored certain customer types for reasons unrelated to the value of a customer type.

We address this possibility by embedding a probabilistic model into our augmented recommender system. For each salesperson-customer dyad that has no sales records, we allow for two possibilities: that the salesperson has considered the customer type, as modeled in the main analysis, or the salesperson has not considered the customer type so that the missing sales record should be excluded from training. We parameterize the probability that salesperson i has considered a customer type as $1/(1 + e^{-\alpha * \text{Experience}_i})$, where α is a hyperparameter to be tuned through cross-validation and where experience is measured by the number of years the salesperson has worked at the company. This probability enters the neural network as the weight for each data point.

We repeat the analysis of prediction accuracy and recommendation quality in this probabilistic framework. We set the number of recommendations to one for the F1-score and NDCG. Table 6 presents the results. Compared with the augmented recommender system in the main analysis (main model), this extended, more flexible model significantly improves recommendation quality.

Table 6. Extensions to the Augmented Recommender System.

Recommender System	Performance Measure		
	MSE	F1-Score	NDCG
Deep Learning — Augmented (Main Model)	100	0.081	0.385
Deep Learning — Probabilistically Missing Sales (Extension)	99.82	0.083***	0.397***
Deep Learning — Heterogeneous Cost of Selling (Extension)	99.12***	0.087***	0.412**
# Observations	272,138	2,430	2,430

Notes. For MSE, each observation is a salesperson-customer dyad in the test data. For the F1-score and NDCG, each observation is a salesperson in the test data. Significance pertains to comparison with the main model. *** $p < 0.01$, ** $p < 0.05$.

6.2 Heterogeneous Net Cost of Selling

In the main analysis, we assume a homogeneous net cost of selling, c, across salespeople. We now relax this assumption. To keep this extension tractable, we assume that there exist two segments of salespeople who differ in their net cost of selling.

We again embed a probabilistic model into our augmented recommender system. We parameterize the probability that salesperson i belongs in one of the two segments as a function of the salesperson's average number of monthly transactions: $1/(1+e^{-\beta * \text{Transactions}_i})$, where β is a hyperparameter to be tuned through cross-validation. As one would expect, the net cost of selling is likely higher for salespeople who accomplish fewer transactions each month. This probability enters training as the weight of each data point in two neural networks separately trained for the two segments.

The bottom row of Table 6 presents the results. Considering heterogeneous net cost of selling leads to significantly better prediction accuracy and recommendation quality. This is expected given the greater flexibility of the extended model.

6.3 Other Possible Extensions

The main analysis and both extensions boil down to one central question – how do we interpret the incidence of sales records? Throughout the paper, we highlight the importance of offering behaviorally meaningful, microfounded interpretations of this data feature. The same idea can be applied to potentially many other extensions. For example, a company may take charge of customer prospecting and then assign different prospects to different salespeople. This is the telemarketing setting studied in Gong et al. (2021). In this case, a salesperson's sales records with a customer type may be a function of the company's job assignment rule. Our method can again be easily extended to this scenario by weighing each salesperson-customer dyad with the possibility of encounter.

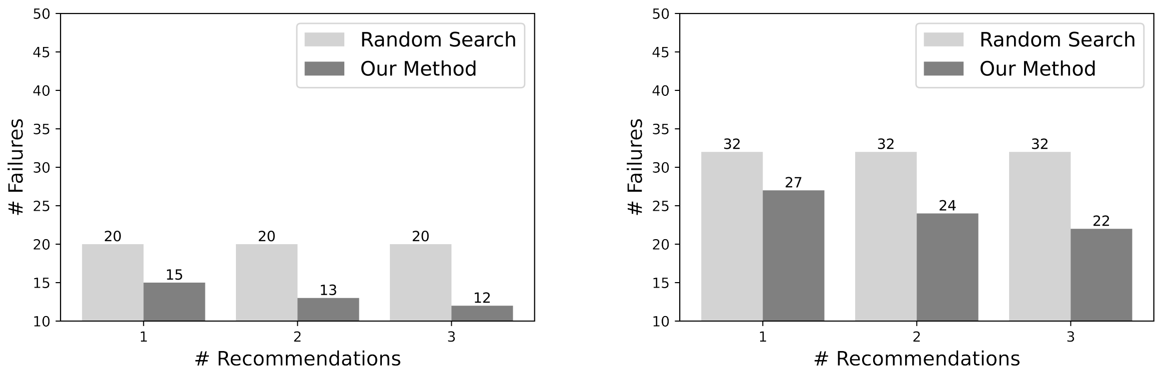
7 Practical Value

We return to the practical motivation of our augmented recommender system. We simulate whether our recommender system indeed helps new salespeople make their first sale and helps the company improve sales force management.

For each salesperson in the test sample, we examine two counterfactual scenarios. We assume that the salesperson approaches customers following either random search or the recommender system. Random search is a simplification but nevertheless plausible strategy if a new salesperson has no guidance, consistent with our interview with the company. With random search, the salesperson randomly chooses a customer type at each attempt to sell. With a recommender system, we assume that the salesperson first tries the recommended customer types and, if doing so does not generate a sale, goes on to approach other customers through random search. For either approach, if we observe a sale between the salesperson and the chosen customer type in the company's historical sales records, we count the sales attempt as a success; otherwise, we count it as a failure.

Figure 6a presents the average number of failures before the first sale for random search and our method, respectively. We vary the number of recommendations from one to three. As expected, our method outperforms random search more if it is allowed to recommend more customer types. For instance, by recommending three customer types, our method reduces the number of failures from 20 to 12, a 40% improvement, compared with random search. We present the comparison with other benchmark recommender system in the Online Appendix, where our model continues to outperform the benchmarks.

Figure 6. Number of Failures before the First Sale.



(a) All Salespeople (b) New Salespeople

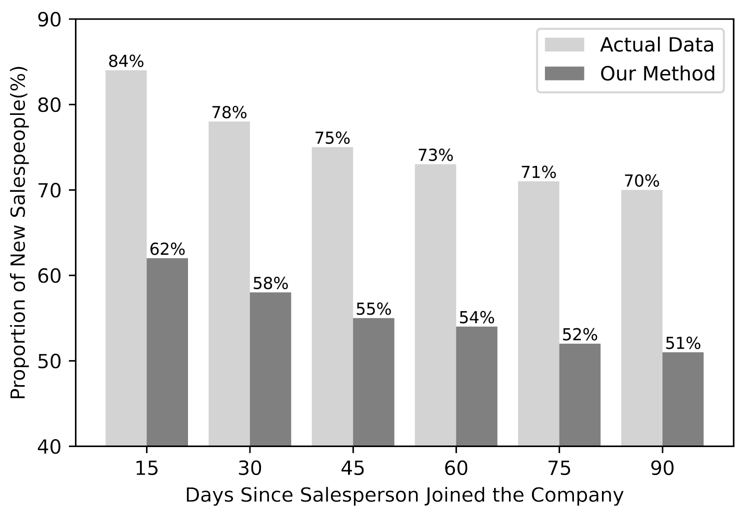
We look further at how our recommender system can help new salespeople specifically. We focus on the subsample of salespeople in the test data who have worked at the company for less than one year. We redo the above analysis and Figure 6b presents the results. Indeed, new salespeople tend to go through more failures before their first sale. Our recommender system can help them significantly expedite their first success.

Last, we simulate the aggregate implication of expediting sales for the company. Recall Figure 3, which presents the proportion of new salespeople yet to make their first sale as time goes by. We again assume that new salespeople follow random search when not aided by a recommender system. In addition, we perform a back-of-envelope calculation of new salespeople's pace of sales attempts, so that the fraction of salespeople who have not achieved the first sale over time fits the actual data in Figure 3.[19] We then use the calculated pace of sales attempts to simulate the number of salespeople yet to make their first sale over time when a recommender system is in place.

[19]The number of days a salesperson takes to make the first sale equals the number of failures before the first sale multiplied by the number of days a salesperson takes to make a sales attempt (the pace). We allow the pace of making a sales attempt to be time-varying. There are six 15-day intervals in Figure 3. Thus we calculate six corresponding paces based on the proportion of salespeople who have not made a sale by that time. The results are 2.83, 4.76, 6.16, 7.69, 9.04, and 10.47 days per sales attempt for these six intervals, respectively. The pace slows down possibly due to frustration.

We present the full results for different recommender systems and different numbers of recommendations in the Online Appendix. Figure 7 displays the result of using our augmented recommender system to recommend only one customer type. Even in this most conservative case, our method greatly improves sales force productivity, reducing the number of salespeople with no sales by 27% 90 days into their job. Expediting the first sale can accelerate revenue and reduces attribution. It may also generate positive ripple effects through better customer satisfaction and sales force morale.

Figure 7. Proportion of New Salespeople Who Have Not Made Their First Sale, Revisited.



8 Concluding Remarks

In this paper, we develop a deep learning based recommender system to help new salespeople recognize customers with high conversion potential. We augment standard deep learning based recommender systems to capture an important feature of the sales environment, that experienced salespeople's failures to convert certain customer types is as informative as their success. We develop a parsimonious behavioral model to capture what we refer to as "missing by choice" sales records in company databases. We then incorporate the model into a neural network structure by proposing a new activation function. Our augmented recommender system outperforms common benchmarks in prediction accuracy and recommendation quality. Simulation suggests that our method can markedly improve sales prospecting efficiency and sales force productivity.

The contributions of this paper are two-fold. Substantively, we develop a data-driven, automated system to improve performance in the sales function, which is known for its intense pressure and high stakes. Methodologically, we develop a way to augment already-powerful deep learning based recommender systems to handle missing-by-choice observations. Our augmentation performs well, and is simple, interpretable, and easily extendable. We showcase its efficacy in the sales context, but we expect the framework to be broadly applicable. Again, take the classic example of customer reviews. Consumers may choose not to review a product on a website if they feel they have no new opinions to add (Chakraborty et al. 2021). This will lead to missing-by-choice reviews. An augmented deep learning based recommender system can be applied to retrieve the opinions these consumers may have had, considering the content of existing reviews (e.g., Toubia et al. 2021) or quality signals from existing product images (Zhang et al. 2021).

There are many avenues for future research. First, it will be useful to evaluate the impact of a recommended system like ours on sales performance in the field. Various factors need to be unpacked, such as the mere effect of introducing artificial intelligence into a traditional industry, which can be complex in itself. Second, by recommending similar customer types to similar salespeople, the recommender system may intensify competition. Competition is not a salient concern in our setting because the potential market of customers is enormous and widely dispersed. It is nevertheless an interesting topic to explore. Last, as a proof of concept, our recommender system abstracts away from many rich features of the sales environment, such as salesperson learning and team dynamics (Lim and Chen 2014). Incorporating these features can unlock valuable opportunities to further improve recommender system design.

References

Aggarwal, C. C. (2016). *Recommender Systems: The Textbook*. SpringerLink.

Ahearne, M. (2019). Rethinking marketing. Technical report, University of Houston.

Altmann, A., L. Tolosi, O. Sander, and T. Lengauer (2010). Permutation importance: A corrected feature importance measure. *Bioinformatics 26*(10), 1340–1347.

Ansari, A., S. Essegaier, and R. Kohli (2000). Internet recommendation systems. *Journal of Marketing Research 37*(3), 363–375.

Ansari, A., Y. Li, and J. Z. Zhang (2018). Probabilistic topic model for hybrid recommender systems: A stochastic variational bayesian approach. *Marketing Science 37*(6), 987–1008.

Athey, S. and G. W. Imbens (2019). Machine learning methods that economists should know about. *Annual Review of Economics 11*, 685–725.

Bobadilla, J., F. Ortega, A. Hernando, and J. Bernal (2012). A collaborative filtering approach to mitigate the new user cold start problem. *Knowledge-Based Systems 26*, 225–238.

Bodapati, A. V. (2008). Recommendation systems with purchase data. *Journal of marketing research 45*(1), 77–93.

Chakraborty, I., J. Deb, and A. Öry (2021). When do consumers talk? Working paper.

Chang, L. J., B. B.Doll, M. van't Wout, M. J. Frank, and A. G. Sanfey (2010). Seeing is believing: Trustworthiness as a dynamic belief. *Cognitive Psychology 61*(2), 87–105.

Chen, H. and N. Lim (2017). How does team composition affect effort in contests? A theoretical and experimental analysis. *Journal of Marketing Research 54*(1), 44–60.

Chen, J., H. Dong, X. Wang, F. Feng, M. Wang, and X. He (2021). Bias and debias in recommender system: A survey and future directions. *arXiv preprint arXiv:2010.03240*.

Cheng, H.-T., L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir, et al. (2016). Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*, pp. 7–10.

Chintagunta, P. K. (1993). Investigating purchase incidence, brand choice and purchase quantity decisions of households. *Marketing Science 12*(2), 184–208.

Chu, W.-T. and Y.-L. Tsai (2017). A hybrid recommendation system considering visual information for predicting favorite restaurants. *World Wide Web 20*, 1313–1331.

Chung, D. J., B. Kim, and N. B. Syam (2020). A practical approach to sales compensation: What do we know now? What should we know in the future? *Foundations and Trends in Marketing 14*(1), 1–52.

Chung, D. J., T. Steenburgh, and K. Sudhir (2014). Do bonuses enhance sales productivity? A dynamic structural analysis of bonus-based compensation plans. *Marketing Science 33*(2), 165–187.

Chung, J. and V. R. Rao (2012). A general consumer preference model for experience products: Application to internet recommendation services. *Journal of Marketing Research 49*(3), 289–305.

Chung, T. S., R. T. Rust, and M. Wedel (2009). My mobile music: An adaptive personalization system for digital audio players. *Marketing Science 28*(1), 52–68.

Covington, P., J. Adams, and E. Sargin (2016). Deep neural networks for YouTube recommendations. In *Proceedings of the 10th ACM conference on recommender systems*, pp. 191–198.

Cui, P., X. Wang, J. Pei, and W. Zhu (2019). A survey on network embedding. *IEEE Transactions on Knowledge and Data Engineering 31*(5), 833–852.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems 2*, 303–314.

Daljord, Ø., S. Misra, and H. S. Nair (2016). Homogeneous contracts for heterogeneous agents: Aligning sales force composition and compensation. *Journal of Marketing Research 53*(2), 161–182.

Devlin, J., M.-W. Chang, K. Lee, and K. Toutanova (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4171–4186.

Dew, R. (2021). Preference measurement with unstructured data, with applications to adaptive onboarding surveys. Working paper.

Donnelly, R., A. Kanodia, and I. Morozov (2022). Welfare effects of personalized rankings. Working paper.

Duarte, J., S. Siegel, and L. Young (2012). Trust and credit: The role of appearance in peer-to-peer lending. *Review of Financial Studies 25*(8), 2455–2484.

Dube, J.-P. and S. Misra (2019). Personalized pricing and customer welfare. Working paper.

Dziugaite, G. K. and D. M. Roy (2015). Neural network matrix factorization. *arXiv preprint arXiv:1511.06443*.

Dzyabura, D. and J. R. Hauser (2019). Recommending products when consumers learn their preference weights. *Marketing Science 38*(3), 417–441.

Findlay, D. (1989). Training networks with discontinuous activation functions. In *1989 First IEE International Conference on Artificial Neural Networks*.

Franco, B. (2015). *Sales: The Hardest Easy Job in the World*. Cushing-Malloy Books.

Gabel, S. and A. Timoshenko (2021). Product choice with large assortments: A scalable deep-learning model. *Management Science*.

Ganesan, S. and R. Hess (1997). Dimensions and levels of trust: Implications for commitment to a relationship. *Marketing Letters 8*(4), 439–448.

Gardete, P. M. and C. D. Santos (2020). No data? No problem! A search-based recommendation system with cold starts. Working paper.

Gheorghiu, A. I., M. J. Callan, and W. J. Skylark (2017). Facial appearance affects science communication. *PNAS 114*(23), 5970–5975.

Godin, F., J. Degrave, J. Dambre, and W. D. Neve (2018). Dual rectified linear units (DReLUs): A replacement for Tanh activation functions in quasi-recurrent neural networks. *Pattern Recognition Letters 16*, 8–14.

Gomez-Uribe, C. A. and N. Hunt (2015). The netflix recommender system: Algorithms, business value, and innovation. *ACM Transactions on Management Information Systems 6*(4), 1–19.

Gong, S., Q. Li, S. Su, and J. Zhang (2021). Genes and sales. Working paper.

Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep learning*. MIT Press Cambridge.

Gower, S. (2014). Netflix prize and SVD. Notes.

Graves, A., A. rahman Mohamed, and G. Hinton (2013). Speech recognition with deep recurrent neural networks. In *International Conference on Acoustics, Speech, and Signal Processing*.

Hartmann, J., M. Heitmann, C. Schamp, and O. Netzer (2021). The power of brand selfies. *Journal of Marketing Research 58*(6), 1159–1177.

Hauser, J. R., G. L. Urban, G. Liberali, and M. Braun (2009). Website morhping. *Marketing Science 28*(2), 202–223.

Hedges, N. (2016). How to deal with stress in sales. *Inc.*, November 23.

Hernandez-Lobato, J. M., N. Houlsby, and Z. Ghahramani (2014). Probabilistic matrix factorization with non-random missing data. In *ICML'14: Proceedings of the 31st International Conference on Machine Learning*, pp. 1512–1520.

Horowitz, J. L. (2001). The bootstrap. In J. J. Heckman and E. Leamer (Eds.), *Handbook of Econometrics*, pp. 3160–3228. Elsevier.

Huang, D. and L. Luo (2016). Consumer preference elicitation of complex products using fuzzy support vector machine active learning. *Marketing Science 35*(3), 445–464.

Koren, Y., R. Bell, and C. Volinsky (2009). Matrix factorization techniques for recommender systems. *IEEE Computer Society 42*(8), 30–37.

Laffont, J.-J. and D. Martimort (2009). *The Theory of Incentives: The Principal-Agent Model*. Princeton University Press.

Lau, M. M. and K. H. Lim (2018). Review of adaptive activation function in deep neural network. In *IEEE EMBS Conference on Biomedical Engineering and Sciences*.

Lieven, T. (2016). Customers' choice of a salesperson during the initial sales encounter. *Journal of Retailing and Consumer Services 32*, 109–116.

Lim, N. and H. Chen (2014). When do group incentives for salespeople work? *Journal of Marketing Research 51*(3), 320–334.

Loewenstein, G., T. O'Donoghue, and M. Rabin (2003). Projection bias in predicting future utility. *The Quarterly Journal of Economics 118*(4), 1209–1248.

Lu, J., D. Wu, M. Mao, W. Wang, and G. Zhang (2015). Recommender system application developments: A survey. *Decision Support Systems 74*, 12–32.

Luo, X., M. S. Qin, Z. Fang, and Z. Qu (2021). Artificial intelligence coaches for sales agents: Caveats and solutions. *Journal of Marketing 85*(2), 14–32.

Malik, N., P. V. Singh, and K. Srinivasan (2020). A dynamic analysis of attractiveness premium. Working paper.

Marlin, B., R. S. Zemel, S. Roweis, and M. Slaney (2007). Collaborative filtering and the missing at random assumption. In *Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence*, pp. 267–275.

Misra, S. (2019). Selling and sales management. In J.-P. Dube and P. Rossi (Eds.), *Handbook of the Economics of Marketing*, pp. 441–491. Elsevier.

Misra, S. and H. S. Nair (2011). A structural model of sales-force compensation dynamics: Estimation and field implementation. *Quantitative Marketing and Economics 9*, 211–257.

Mullins, R. R., M. Ahearne, S. K. Lam, Z. R. Hall, and J. P. Boichuk (2014). Know your customer: How salesperson perceptions of customer relationship quality form and influence account profitability. *Journal of Marketing 78*, 38–58.

Najafabadi, M. M., F. Villanustre, T. M. Khoshgoftaar, N. Seliya, R. Wald, and E. Muharemagic (2015). Deep learning applications and challenges in big data analytics. *Journal of Big Data 2*, 1–21.

Padilla, N. and E. Ascarza (2021). Overcoming the cold start problem of CRM using a probabilistic machine learning approach. *Journal of Marketing Research 58*(5), 981–1006.

Proserpio, D., J. R. Hauser, X. Liu, T. Amano, A. Burnap, T. Guo, D. D. Lee, R. Lewis, K. Misra, E. Schwarz, A. Timoshenko, L. Xu, and H. Yoganarasimhan (2020). Soul and machine (learning). *Marketing Letters 31*, 393–404.

Rabin, M. (2000). Risk aversion and expected-utility theory: A calibration theorem. *Econometrica 68*(5), 1281–1292.

Rajaram, P. and P. Manchanda (2020). Video influencers: Unboxing the mystique. *arXiv preprint arXiv:2012.12311*.

Ricci, F., L. Rokach, and B. Shapira (2015). *Recommender Systems Handbook* (Second ed.). Springer.

Richard, O. C., M. M. Stewart, P. F. McKay, and T. W. Sackett (2017). The impact of store-unit-community racial diversity congruence on store-unit sales performance. *Journal of Management 43*(7), 2386–2403.

Schnabel, T., A. Swaminathan, A. Singh, N. Chandak, and T. Joachims (2016). Recommendations as treatments: Debiasing learning and evaluation. In *ICML'16: Proceedings of the 33rd International Conference on Machine Learning*, pp. 1670–1679.

Silverira, T., M. Zhang, X. Lin, Y. Liu, and S. Ma (2019). How good your recommender system is? A survey on evaluations in recommendation. *International Journal of Machine Learning and Cybernetics 10*, 813–831.

Simester, D., A. Timoshenko, and S. I. Zoumpoulis (2020). Targeting prospective customers: Robustness of machine-learning methods to typical data challenges. *Management Science 66*(6), 2495–2522.

Sofer, C., R. Dotsch, D. H. Wigboldus, and A. Todorov (2015). What is typical is good: The influence of face typicality on perceived trustworthiness. *Psychological Science 26*(1), 39–47.

Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research 15*(56), 1929–1958.

Steck, H. (2013). Evaluation of recommendations: Rating-prediction and ranking. In *Proceedings of the 7th ACM conference on Recommender systems*, pp. 213–220.

Stirrat, M. and D. Perrett (2010). Valid facial cues to cooperation and trust: Male facial width and trustworthiness. *Psychological Science 21*(3), 349–354.

Stoker, J. I., H. Garretsen, and L. J. Spreeuwers (2016). The facial appearance of ceos: Faces signal selection but not performance. *PLoS ONE 11*(7).

Sunder, S., V. Kumar, A. Goreczny, and T. Maurer (2017). Why do salespeople quit? An empirical examination of own and peer effects on salesperson turnover behavior. *Journal of Marketing Research 54*(3), 381–397.

Thaler, R. H. (2008). Mental accounting and consumer choice. *Marketing Science 27*(1), 15–25.

Tkachenko, Y. and K. Jedidi (2020). What personal information can a consumer facial image reveal? Implications for marketing roi and conusmer privacy. Working paper.

Tobin, J. (1958). Estimation of relationships for limited dependent variables. *Econometrica 26*(1), 24–36.

Toubia, O., J. Berger, and J. Eliashberg (2021). How quantifying the shape of stories predicts their success. *Proceedings of the National Academy of Sciences 118*(26).

Urban, G. L., J. R. Hauser, G. Libertali, M. Braun, and F. Sultan (2009). Morph the web to build empathy, trust and sales. *MIT Sloan Management Review 50*(4), 52–61.

Urban, G. L., F. Sultan, and W. J. Qualls (2000). Placing trust at the center of your internet strategy. *MIT Sloan Management Review 42*(4), 39–48.

Wang, H. and D.-Y. Yeung (2020). A survey on bayesian deep learning. *ACM Computing Survey 53*(5), 1–37.

Wang, S., Y. Wang, J. Tang, K. Shu, S. Ranganath, and H. Liu (2017). What your images reveal: Exploiting visual contents for point-of-interest recommendation. In *WWW'17: Proceedings of the 26th International Conference on World Wide Web*, pp. 391–400.

Wei, J., J. He, K. Chen, Y. Zhou, and Z. Tang (2017). Collaborative filtering and deep learning based recommendation system for cold start items. *Expert Systems With Applications 69*(1), 29–39.

Williams, K. C. and R. L. Spiro (1985). Communication style in the salesperson-customer dyad. *Journal of Marketing Research 22*(4), 434–442.

Ying, Y., F. Feinberg, and M. Wedel (2006). Leveraging missing ratings to improve online recommendation systems. *Journal of Marketing Research 43*(3), 355–365.

Yoganarasimhan, H. (2020). Search personalization using machine learning. *Management Science 66*(3), 1045–1070.

Zhang, C., S. Bengio, M. Hardt, B. Recht, and O. Vinyals (2021). Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM 64*(3), 107–115.

Zhang, S., D. Lee, P. V. Singh, and K. Srinivasan (2021). What makes a good image? Airbnb demand analytics leveraging interpretable image features. *Management Science*.

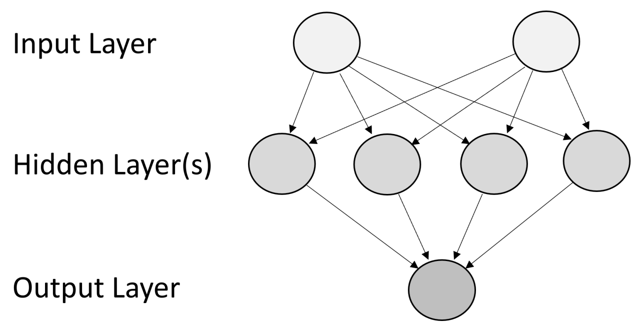
Zhang, S., L. Yao, A. Sun, and Y. Tay (2019). Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys 52*(1), 1–38.

Zhang, Y. and X. Chen (2020). Explainable recommendation: A survey and new perspectives. *Foundations and Trends in Information Retrieval 14*(1), 1–101.

Online Appendix

A.1 Neural Network Illustration

Figure A1. A Typical Neural Network Structure.



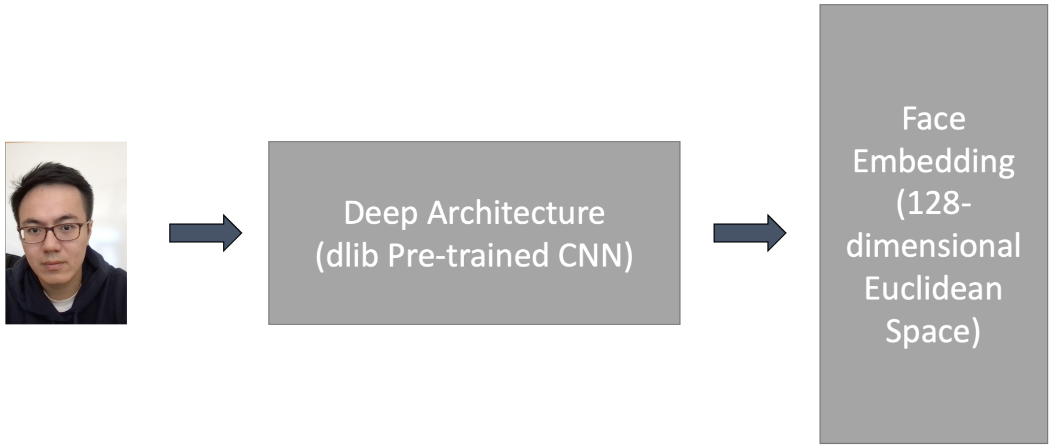
A.2 Details on the Augmented Recommender System

We implement the augmented recommender system using the Keras functional API. The main components of the hidden layers in our augmented recommender system are the embedding layer and the MLP.

The embedding of the categorical features is implemented through Keras Embedding and the output embedding dimension is set as $\min\{50, m/2\}$, where m is the number of categories per feature (Howard and Gugger 2020). We derive the face embeddings using the dlib library.[A1] The output is a 128-dimensional vector for each face. The pre-trained network is a ResNet network with 29 convolutional layers, which is based on He et al. (2016) with fewer layers and filters. The network is trained on the Labeled Faces in the Wild (LFW) dataset, which includes around 3 million images.[A2] We illustrate the derivation of face embeddings in Figure A2. The embedding vector itself does not carry interpretable information. But similar faces have smaller distances in the embedding space.

[A1] http://blog.dlib.net/2017/02/high-quality-face-recognition-with-deep.html.
[A2] http://vis-www.cs.umass.edu/lfw.

Figure A2. Illustration of Face Embedding Derivation

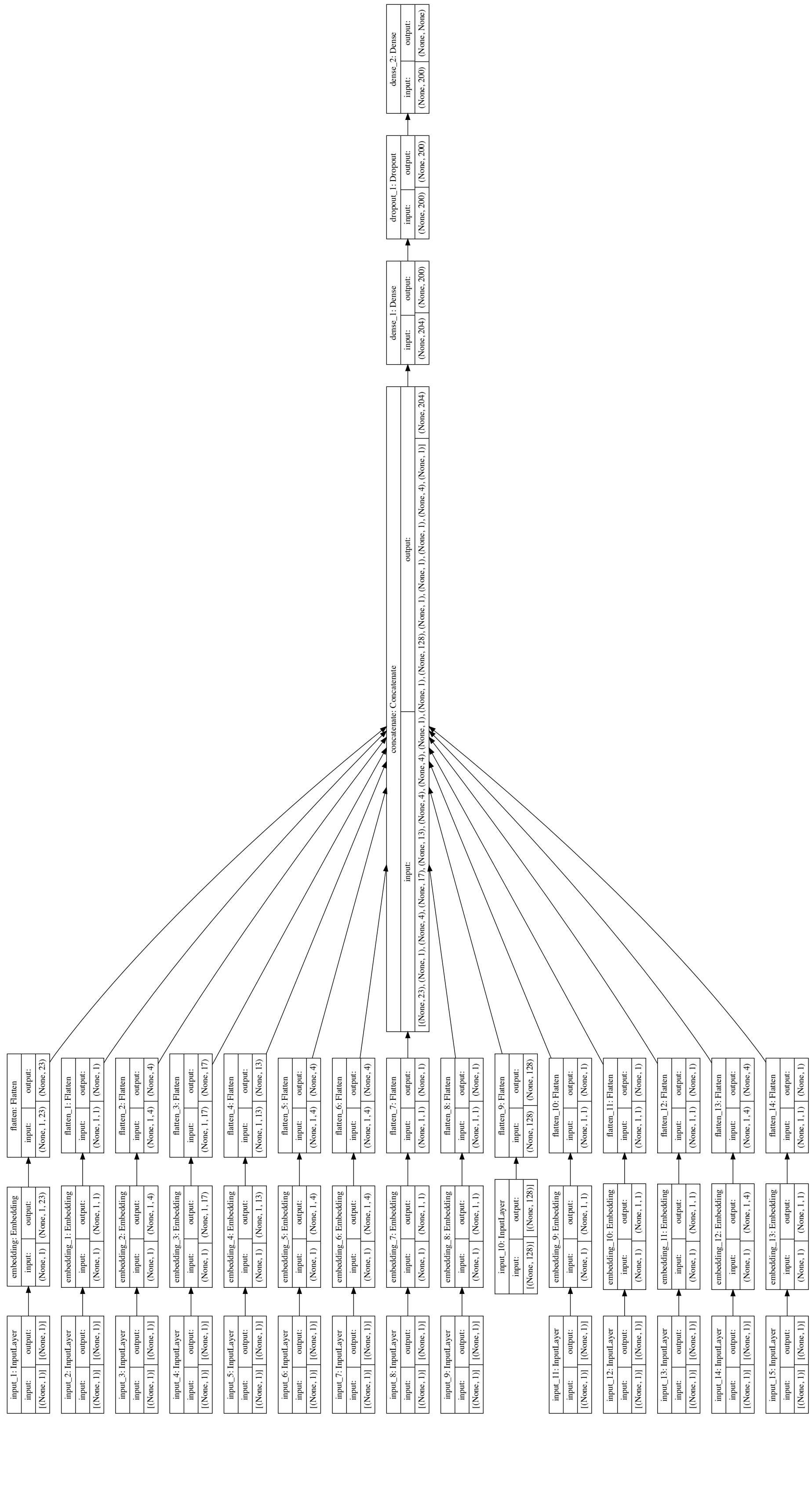


Our MLP includes the now-common ReLU activation function in each layer (Farrell et al. 2021). We also use Dropout at each layer to prevent neural networks from overfitting and set the fraction of dropout at 0.5 (Srivastava et al. 2014). We use stochastic gradient descent and early stopping (Prechelt 2002) to obtain the number of epochs. The number of hidden layers is set at two (Zhang et al. 2021), while the number of units in each layer is derived through cross-validation.

For the “Deep Learning — Missing Excluded” benchmark, we minimize its validation loss (MSE) on the validation data that have positive sales records. This replicates the construction of the optimal neural network underlying this benchmark.

Figure A3 summarizes the neural network structure of our augmented recommender system, the “Deep Learning — Augmented” model. The tenth input in Figure A3 is our facial embeddings derived using the dlib library.

Figure A3. Neural Network Structure of the Augmented Deep Learning Based Recommender System.



While using the “Deep Learning — Augmented” method, we also need to fine-tune the net cost of selling, c. We first obtain the lowest and highest predicted values of sales revenue generated from the MLP. We evenly split the interval between these two values to ten blocks and use the eleven endpoints as starting values of c to train the neural network. We obtain the c that generates the smallest MSE on the validation data. We then take the two endpoints surrounding this c, use them to define a new interval, evenly split this new interval to ten blocks, and use the eleven new endpoints as new values of c to train the neural network. We iterate this process until convergence, defined as the MSE on the validation data ceasing to decrease or the optimal c ceasing to change (difference smaller than 0.00001).

Figures A4, A5, and A6 respectively summarize the training and validation loss across epochs for the “Deep Learning — Missing Excluded,” “Deep Learning — Linear Activation,” and “Deep Learning — Augmented” methods. We scale the training loss at epoch 1 to 100, following our confidentiality agreement with the company.

Figure A4. Training and Validation Loss: Missing Values Excluded.

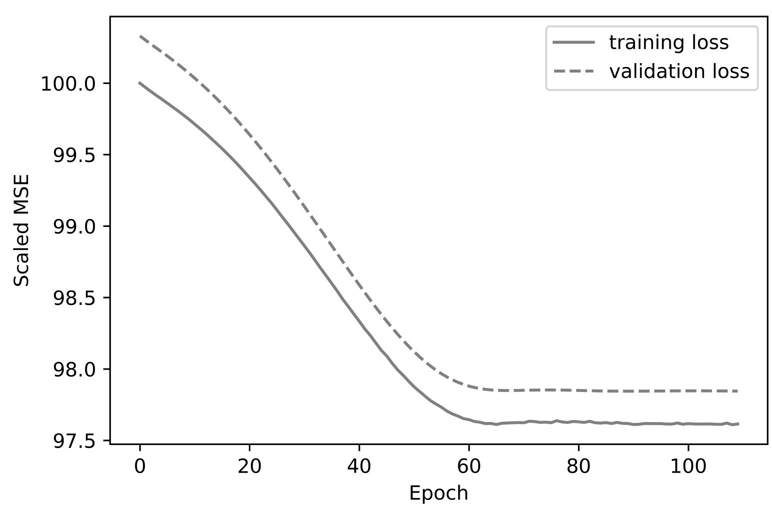


Figure A5. Training and Validation Loss: Linear Activation.

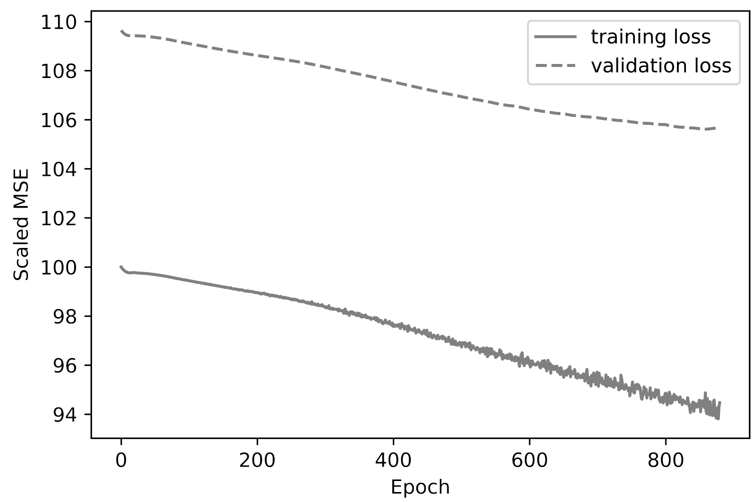
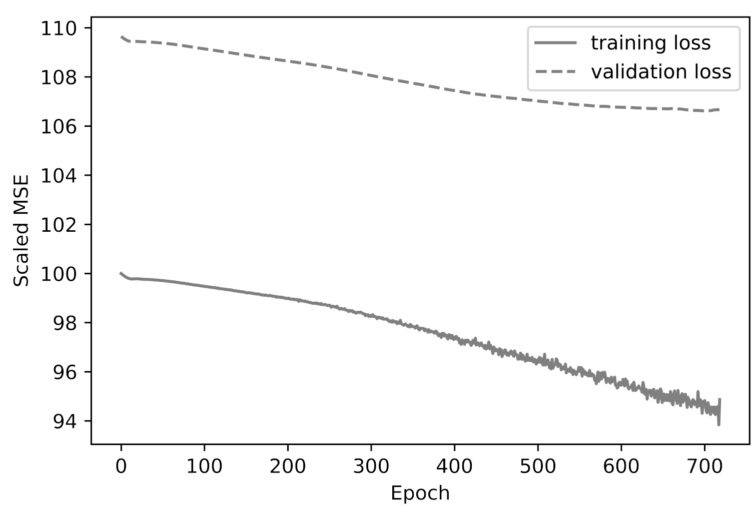
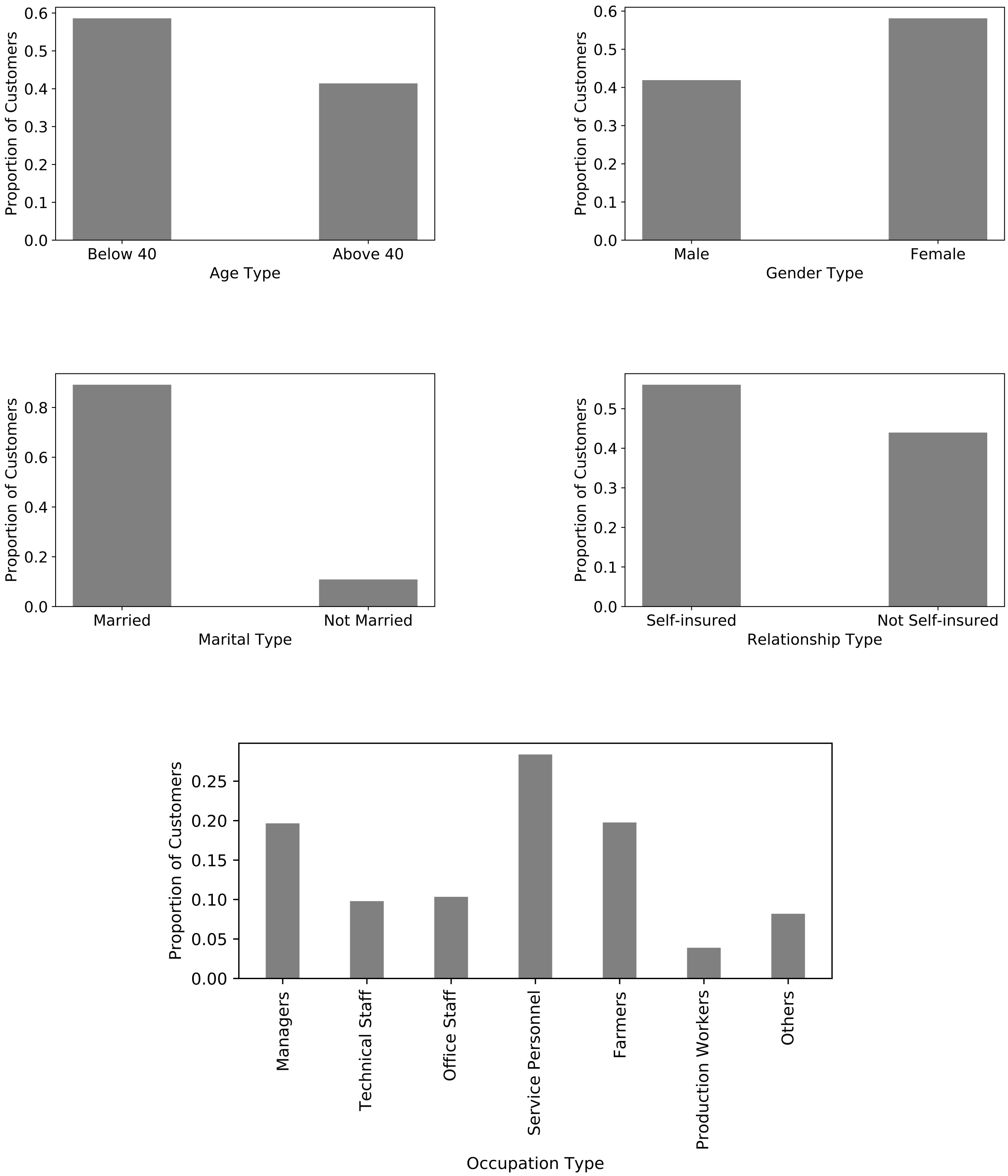


Figure A6. Training and Validation Loss: Augmented.



A.3 Distribution of Customer Types for Each Customer Trait

Figure A7. Distribution of Customer Types for Each Customer Trait.



A.4 Recommendation Quality by Salespeople Experience When More Customer Types Are Recommended

Table A1. Recommendation Quality by Salesperson Experience (Multiple Customer Types Recommended).

Recommender System	Years Worked at Company		
	0	1	2-6
When Two Customer Types Are Recommended			
		F1-Score	
Non-Personalized Recommender System	0.072***	0.076***	0.076***
Deep Learning — Missing Excluded	0.038***	0.043***	0.051***
Deep Learning — Linear Activation	0.119***	0.122***	0.124***
Deep Learning — Augmented (Our Method)	0.127	0.138	0.131
		NDCG	
Non-Personalized Recommender System	0.179***	0.273***	0.434***
Deep Learning — Missing Excluded	0.095***	0.150***	0.296***
Deep Learning — Linear Activation	0.290***	0.390***	0.639***
Deep Learning — Augmented (Our Method)	0.299	0.404	0.646
When Three Customer Types Are Recommended			
		F1-Score	
Non-Personalized Recommender System	0.093***	0.105***	0.115***
Deep Learning — Missing Excluded	0.046***	0.056***	0.069***
Deep Learning — Linear Activation	0.142***	0.154***	0.167***
Deep Learning — Augmented (Our Method)	0.147	0.170	0.174
		NDCG	
Non-Personalized Recommender System	0.223***	0.321***	0.478***
Deep Learning — Missing Excluded	0.117***	0.177***	0.329***
Deep Learning — Linear Activation	0.332***	0.429***	0.648***
Deep Learning — Augmented (Our Method)	0.336	0.442	0.657
# Observations	671	884	875

Notes. Each observation is a salesperson in the test data. Significance pertains to comparison with our method. *** $p < 0.01$.

A.5 Recommended Customer Type by Salesperson Trait

We present a simplified illustration of the output of our augmented deep learning based recommender system. The actual output is at the individual-salesperson level. To simplify presentation, we group salespeople based on their traits. We divide each of the 10 salesperson traits into two levels. Specifically, for the non-binary traits, we divide salespeople's age when joining the company to below or weakly above 40, education to below or weakly above college, title in the company to entry level or above, and years working at company to weakly below or above one year. For salespeople's home province and branch served, we focus on the two most frequent types for either. These are Hebei and Jiangxi for salespeople's home province and Hebei and Jiangsu for branch served. Last, for facial image, we split the salespeople into two groups using K-means clustering based on their facial embedding vector values. This yields 20 scenarios (10 salesperson traits and two levels per trait). For each of the 20 scenarios, we output the most recommended customer type using our augmented recommender system. Table A2 presents the results.

Table A2. Most Recommended Customer Type by Salesperson Trait.

Salesperson Trait	Trait Level	Most Recommended Customer Type
Facial Image	Group 0	Above 40 + Female + Married + Service personnel + Self-insured
	Group 1	Below 40 + Male + Married + Managers + Self-insured
Age (When Joining Company)	Below 40	Below 40 + Male + Married + Managers + Self-insured
	Above 40	Above 40 + Female + Married + Service personnel + Self-insured
Gender	Male	Below 40 + Male + Married + Managers + Self-insured
	Female	Below 40 + Female + Married + Service personnel + Self-insured
Education	Below College	Above 40 + Female + Married + Service personnel + Self-insured
	Above College	Below 40 + Male + Married + Managers + Self-insured
Home Province	Hebei	Below 40 + Female + Married + Farmers + Other-insured
	Jiangxi	Above 40 + Female + Married + Service personnel + Self-insured
Branch Served	Hebei	Below 40 + Female + Married + Farmers + Other-insured
	Jiangsu	Below 40 + Female + Married + Managers + Self-insured
Title in Company	Entry Level	Below 40 + Male + Married + Managers + Self-insured
	Above Entry Level	Below 40 + Female + Married + Managers + Self-insured
Years Worked at Company	Below One Year	Below 40 + Male + Married + Managers + Self-insured
	Above One Year	Above 40 + Female + Married + Service personnel + Self-insured
Whether Referred to Join	Not Referred	Below 40 + Female + Married + Managers + Self-insured
	Referred	Below 40 + Female + Married + Managers + Self-insured
Whether Left the Company	Not Left	Below 40 + Female + Married + Service personnel + Self-insured
	Left	Below 40 + Male + Married + Managers + Self-insured

A.6 Number of Failures before the First Sale

In this section, we present the number of failures before the first sale if the salesperson follows a recommender systems. We allow each recommender system to recommend one to three customer types. Table A3 presents the results for all salespeople and for new salespeople, respectively. In all cases, our method outperforms all benchmark recommender systems significantly ($p < 0.01$).

Table A3. Number of Failures before the First Sale.

Recommender System	# Recommendations		
	1	2	3
For All Salespeople			
Random Search	19.65***	19.65***	19.65***
Non-Personalized Recommender System	17.78***	16.72***	15.31***
Deep Learning — Missing Excluded	19.01***	18.72***	18.47***
Deep Learning — Linear Activation	15.91***	13.79***	12.55***
Deep Learning — Augmented (Our Method)	15.20	13.27	11.97
For New Salespeople			
Random Search	31.76***	31.76***	31.76***
Non-Personalized Recommender System	29.61***	28.14***	26.24***
Deep Learning — Missing Excluded	31.22***	30.71***	30.46***
Deep Learning — Linear Activation	27.46***	24.43***	22.73***
Deep Learning — Augmented (Our Method)	26.58	23.81	22.00
# Observations	2,430	2,430	2,430

Notes. Each observation is a salesperson in the test data. Significance pertains to comparison with our method. *** $p < 0.01$.

A.7 Proportion of New Salespeople Yet to Make Their First Sale

Figure 3 of the paper presents the proportion of new salespeople who have not made their first sale as time goes by. We examine how a recommender system changes this proportion. For each recommender system, we vary the number of recommended customer types from one to three. Section 7 of the paper describes the calculation in detail. We use block bootstrapping to test the statistical significance of differences between recommender systems. Table A4 presents the results. In all cases, our method outperforms all benchmark recommender systems significantly ($p < 0.01$).

Table A4. Proportion of New Salespeople Who Have Not Made Their First Sale.

Recommender System	Days Since Salesperson Joined Company					
	15	30	45	60	75	90
	One Customer Type Recommended					
Random Search	0.84***	0.78***	0.75***	0.73***	0.71***	0.70***
Non-Personalized Recommender System	0.71***	0.67***	0.65***	0.64***	0.62***	0.61***
Deep Learning — Missing Excluded	0.81***	0.75***	0.72***	0.71***	0.68***	0.67***
Deep Learning — Linear Activation	0.63***	0.59***	0.56***	0.55***	0.53***	0.52***
Deep Learning — Augmented (Our Method)	0.62	0.58	0.55	0.54	0.52	0.51
	Two Customer Types Recommended					
Random Search	0.84***	0.78***	0.75***	0.73***	0.71***	0.70***
Non-Personalized Recommender System	0.68***	0.64***	0.62***	0.60***	0.59***	0.58***
Deep Learning — Missing Excluded	0.81***	0.74***	0.71***	0.70***	0.67***	0.66***
Deep Learning — Linear Activation	0.55***	0.50***	0.48***	0.47***	0.46***	0.45***
Deep Learning — Augmented (Our Method)	0.52	0.49	0.47	0.46	0.45	0.45
	Three Customer Types Recommended					
Random Search	0.84***	0.78***	0.75***	0.73***	0.71***	0.70***
Non-Personalized Recommender System	0.61***	0.57***	0.56***	0.54***	0.53***	0.52***
Deep Learning — Missing Excluded	0.81***	0.74***	0.70***	0.70***	0.67***	0.65***
Deep Learning — Linear Activation	0.49***	0.46***	0.44***	0.43***	0.42***	0.41***
Deep Learning — Augmented (Our Method)	0.46	0.43	0.42	0.41	0.40	0.40
# Observations	2,430	2,430	2,430	2,430	2,430	2,430

Notes. Each observation is a salesperson in the test data. Significance pertains to comparison with our method. *** $p < 0.01$.

Online Appendix References

Farrell, M. H., T. Liang, and S. Misra (2021). Deep neural networks for estimation and inference. *Econometrica 89*, 181–213.

He, K., X. Zhang, S. Ren, and J. Sun (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition*.

Howard, J. and S. Gugger (2020). *Deep learning for Coders with Fastai and PyTorch: AI Applications Without a PhD* (First ed.). O'Reilly Media.

Prechelt, L. (2002). Early stopping - but when? In G. B. Orr and K.-R. Muller (Eds.), *Neural Networks: Tricks of The Trade*, pp. 55–69. Springer.

Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research 15*(56), 1929–1958.

Zhang, C., S. Bengio, M. Hardt, B. Recht, and O. Vinyals (2021). Understanding deep learning (still) requires rethinking generalization. *Communication of the ACM 64*(3), 107–115.