**[Agent Coaching Recommendation Engine]**

*SDS’s ability to recommend contacts for manual deep dives to operations leaders for targeted opportunity hunting*

Since 2018, SDS-BIA team has launched collection of tools like Agent Scorecard, Balanced Scorecard, Path to Green, DSR Impact Dashboard, utilized by Operations Team to understand performance of CSAs for DSR, PRR, ACHT etc. Contacts from both recipient and driver for all businesses supported by SDS can be visualized at associate level. Few tools normalize for types of contacts handled in different marketplace, channels, routing skills etc. for making apples to apples comparison between two CSAs. A benchmark is calculated using network averages and deviation from this benchmark (whether positive or negative) is used as an indicator of an agent’s performance. Low performing agents are highlighted and team manager/group managers do manual deep dives to understand the causes of low performance.

Knowing the most impactful improvements to delivery success has been a persistent problem within Shipping and Delivery Support and until now, the optimal methods were unknown. Associates utilize resources containing standard operating procedures and are trained on heuristics for common situations, but are still required to make decisions based on imperfect information. The most skilled associates are able to navigate the ambiguity inherent in each customer interaction, and as a result, their actions result in successful deliveries more often than less skilled associates. Historically, the challenge for Shipping and Delivery Support has been in measuring the specific actions influence successful deliveries and instead they have relied on high level correlation analyses. This inability hindered operational efforts to share best practices and eliminate contact handling variation, making it difficult for team managers to evaluate the impact of coaching associates, and preventing program leaders from understanding difference between entitlement and efficacy. As a result, customers were receiving sub-optimal support and packages that could have been delivered, were not.

With agent coaching recommendation engine, customers can be assured that optimal series of actions are taken by every associate of SDS to solve their problems. The feature works by collecting historical data on successfully handled contacts by associates and training a machine learning model to learn scenarios in which majority of the agents achieved successful delivery on package. The scenarios are categorized using permutations of shipment journey that a package witnesses in delivery lifecycle. This is done by taking binary features related to ship-track event scans that happened prior to contact.

“[I am thrilled by this idea and believe it will help my team managers spend their energy finding the right opportunities in the right places - Ashley Pringle]”

Output of the machine learning model is the delivery success probability, which combined to delivery success outcome highlights false positive cases at associate level. After every week, probability success on contacts of prior week is calculated using model that was trained using historical three months data. Team managers manually deep dives randomly chosen false positive contacts by listening them to identify root-cause of the delivery miss. While, mostly the reasons of miss are associate controllable factors, some reasons are uncontrollable like weather issue, protests that have not occurred frequently in the past, such cases are model improvement opportunities. Each deep dive benefits customers because the net result is an increased rate of successful deliveries after a customer support interaction.

“[Quote from a team manager who participated in the pilot of the tool]”

**Frequently Asked Questions**

1. **Where did the idea of [Agent Coaching Recommendation] came from?**

The idea for this project came while designing the shipment path feature for DSR Variance Bridging Dashboard. In contact center, although the allocation of contacts are random, different associates get different scenarios of package journeys and have to improvise on the job to tackle non documented issues using their own skill. The aim of building this recommendation is to highlight contacts handled by associates that are failing in scenarios where majority of the associates are successful. Benchmarking based on volume and metric performance can be misleading, hence a more sophisticated algorithm is required that explains the variance in data distribution of delivery successes as a probability of success.

1. **What can this tool achieve and why now?**

This product improves our ability to solve problems that prevent successful deliveries. After two years since launch of DSR, we are still unable to answer question of what causes variation of DSR performance among different associates and how much quantitative impact it has on our top line metric. This investment would answer the question by transitioning our analysis to be based on actions instead of broad correlations. We would do targeting manual deep dives to identify authentic root cause of failures and ensure each associate goes through training of best practices to deal with ambiguous scenarios while helping our customer receive packages on time. True benchmarking between associates can now be made based on the false positive scenarios, improving the difference in performance across associates.

1. **How should business interpret the output from the machine learning model?**

The primary output of the tool is the delivery success probability which is converted to binary form using a threshold value. Probability >= 0.5 are converted to predicted value of 1 and less than 0.5 are converted to 0. Contacts where the actual delivery success is 0 but predicted value is 1 (probability >= 0.5) are called false positive cases. This implies the model believes based on the performance by majority of associates in the similar scenario, DSR should be 1 instead of 0 implying an opportunity for deep dive to understand the root cause of deviation from expectation. The dashboard highlights false positive contacts at associate level for every team manager to review and deep dive either by manually listening to contacts or identifying the scenario using sic code

1. **How do we define success for this model?**

Quantifying success of the model is challenging because not every recommendation made by the algorithm can be manually deep dived and flagged as accurate vs inaccurate. However, sample estimation technique is used to calculate the model accuracy on weekly basis. Team/Group managers sample false positive contacts and manually mark them as agent controllable vs uncontrollable. Based on this labelling, model success if defined as percentage of false positive contacts that were marked as controllable by different team/group managers.

1. **What are the key features used for building [Agent Coaching Recommendation]?**

All the features considered for building the machine learning model are events that happen prior to contact.

|  |  |  |
| --- | --- | --- |
| **Numerical Features** | lm\_ead\_contact\_diff | Difference between connect datetime and estimate arrival datetime (hours) |
| lm\_wrong\_station\_contact\_diff | Difference between connect datetime and wrong station event datetime (hours) |
| lm\_delay\_contact\_diff | Difference between connect datetime and first delay at station datetime (hours) |
| lm\_missing\_contact\_diff | Difference between connect datetime and first missing datetime (hours) |
| lm\_shipped\_contact\_diff | Difference between connect datetime and first injection datetime (hours) |
| lm\_inducted\_contact\_diff | Difference between connect datetime and first induction datetime (hours) |
| ln\_dispatched\_contact\_diff | Difference between connect datetime and first dispatch datetime (hours) |
| lm\_exception\_contact\_diff | Difference between connect datetime and first delivery attempted time (hours) |
| **Categorical Features** | lm\_ship\_method | ship method of package assigned at the time of ordering |
| channel | mode of communication - email, chat, voice, message us |
| lm\_delivery\_attempted\_reasoncode | First delivery exception reason that happened prior to contact |
| **Binary Features** | lm\_customer\_rejected\_prior\_during\_contact | if customer rejected prior to during the contact |
| lm\_hold\_for\_customer\_request | if package had hold for customer request scan prior to contact |
| lm\_contact\_48h\_prior\_ead | if contact connect datetime was 48 hours prior to EAD |
| lm\_transport\_started\_prior\_contact | if transportation started prior to contact |
| lm\_package\_shipped\_prior\_contact | if package was shipped prior to contact |
| lm\_package\_inducted\_prior\_contact | if package was inducted prior to contact |
| lm\_wrong\_station\_prior\_contact | if package went to wrong station prior to contact |
| lm\_delayed\_at\_station\_prior\_contact | if package had delayed at station scan prior to contact |
| lm\_delayed\_prior\_contact | if package had delayed scan prior to contact |
| lm\_missing\_prior\_contact | if package had missing scan prior to contact |
| lm\_hold\_for\_action\_prior\_contact | is package has hold for action scan prior to contact |
| lm\_hold\_for\_redelivery\_prior\_contact | if package had hold for redelivery scan prior to contact |
| lm\_delivery\_exception\_prior\_contact | if package had delivery exception scan prior to contact |
| lm\_damaged\_prior\_contact | if package had damaged scan prior to contact |
| lm\_undeliverable\_prior\_contact | if package had undeliverable scan prior to contact |
| lm\_dispatched\_prior\_contact | if package was dispatched prior to contact |