“Promotion Classification Model” Module Transcript

Chapter 1

Intro, Topics Covered, & Learning Outcomes

Hey. I'm Hayley, and I'm on the One AI team here at One Model.

From previous modules, you've built a strong foundation of machine learning concepts and model building best practices. Now we'll apply that knowledge by creating a promotion classification model using the One AI recipe. This model will allow you to predict which employees are likely to be promoted and uncover the key factors that influence those promotions, giving you deeper insights into career progression within your organization.

During this module, we will cover an overview of the Promotion Classification Model and some relevant use cases, some key considerations before the model building process, step by step instructions for building a promotion classification model in one AI using the recipe, and we'll close with some helpful insights that can be analyzed after building.

After completing this module, you will understand the basics of a promotion classification model, including its purpose, structure, and key use cases. Identify important considerations before building the model, such as ethical use of predictions and data quality. Gain hands-on experience building a promotion classification model in one AI using the recipe. And, you'll learn how to analyze and interpret the model's insights, uncovering key drivers of promotion outcomes and applying those insights to improve decision-making and strategies.

Chapter 2

Overview & Use Cases

Section 2 - Overview and Use Cases

The promotion analysis model is designed to predict the likelihood of whether an employee will be promoted within a specified time frame, such as next year. By analyzing attributes like performance metrics, organizational and position tenure, skills, as well as manager and team performance, this binary classification model identifies individuals with a high likelihood of being promoted.

The two outcomes that the model will classify employees into are promoted and not promoted. The instances are individual employees, usually identified by a unique ID number from the employee table.

The model creator selects the model population, the prediction period, and the input features used to train the model to make its predictions. Input features include core attributes and individual and team generative attributes.

Additionally, the model will highlight the top factors driving promotions, both for those likely and unlikely to be promoted, giving you deeper insights into what influences career progression in your organization. This data allows you to better align opportunities with employee potential, ensuring you're promoting the right people at the right time while also fostering a culture of growth and development.

Some key use cases for the promotion analysis model include equity and fairness. So the promotion model can highlight disparities by analyzing how different demographic groups, like gender or ethnicity, are classified in terms of promotion likelihood. For instance, if the model consistently predicts lower promotion rates for a specific group, it may reveal bias in the historical data. By identifying these patterns, organizations can adjust promotion criteria or address biases in the decision making process, helping to create more equitable opportunities for all employees.

Next, we have succession planning. So the model can also help identify high potential employees likely to be promoted soon based on factors like performance, skills, and experience. By pinpointing these individuals, organizations can proactively include them in succession planning, ensuring key leadership positions are filled by those already on a growth plan. This enables HR and leadership to focus on developing these employees and preparing them for future roles.

Additionally, recruitment strategy alignment. The model's predictions can reveal which internal candidates are likely to be promoted soon. With this insight, organizations can adjust their recruiting strategy focusing on external hires when internal promotions won't meet the need. By aligning recruitment efforts with these predictions, companies can balance talent development with external hiring as needed.

And finally, the model can serve as a check to ensure the right employees are being promoted by analyzing the key factors driving promotion decisions. By highlighting influences like performance, experience, and skills, organizations can confirm that promotions, both historically and going forward, are merit based rather than influenced by unconscious biases.

Chapter 3

Considerations Before Model Building

Section 3- Considerations Before Model Building

Before creating a promotion classification model in One AI, there are some key factors to think about. First, as with all models built in One AI, the ethical use of predictions is essential. In human resources, predictive models can significantly impact employees' career paths and should be handled with care. The models' outputs should assist, not replace, human decision making and promotions.

Overreliance on AI predictions and critical decisions like promotions can dehumanize the process. To avoid this, when you're defining your objectives and use cases for the model, you can also clearly define how the model's results will be used in conjunction with human judgment to maintain a balance between automation and fairness in promotion decisions.

Next, promotion processes can be influenced by unconscious biases related to gender, race, and tenure, and many other things, which may be embedded in the historical data used to train predictive models. If left unaddressed, the model can replicate these biases leading to unfair outcomes and reinforcing disparities. For instance, if a certain group has been under-promoted in the past, the model might continue predicting lower promotion rates for that group regardless of qualifications. To mitigate this, apply fairness metrics, bias detection techniques, and strategies like disparate impact analysis to reduce bias and promote equitable outcomes across all demographic groups.

Next, let's address the importance of interpretability and explainability. Given the sensitivity of this model, it's essential that human resources teams and managers understand how predictions are made to build trust and encourage informed decision making. A black box approach with no transparency can lead to hesitation in using the model. If building this model in One AI, focus on educating decision makers about the model's results. If using another platform, prioritize explainable AI techniques used in One AI like SHAP values, which clarify how each feature influences promotion likelihood.

And finally, as always, ensure you have accessible, reliable, and maybe most importantly, relevant data to train the model. Verify that the necessary data sources are integrated and validated in one model before beginning.

Next, we'll move over to One Model for a demo on building a promotion classification model in one AI with the recipe.

Chapter 4

How to Build in One AI

Section 4 - Building a Promotion Classification Model in One AI

To get started building your promotion model, click 'One AI' in the main ribbon menu. And this is where you can edit or create models. To add a new model, click 'Add Machine Learning Model' in the upper right hand corner.

Then you'll be prompted to give your model a unique name in the display name field.

Next, you'll click using ‘One AI Recipe’ beneath using data from and 'Configure One AI Recipe'.

And this will bring you to the One AI query builder screen. Under "What are you interested in predicting?", select 'Binary Promotion Analysis' from the recipe list. Once selected, the problem statement is populated with the information we need to provide to build this model.

In step one, we select the column or metric that defines promotions, which is the target value for this model. So what we want the model to predict. First, I'm going to select 'Metric' from this first dropdown because columns should only be used if they were created by your data engineer specifically for One AI purposes.

Next, I'm selecting the promotion metric to be our target variable.

If you do not have a similar metric, you can create one by using the event occurrence column from the employee event table and filtering the Event Reason dimension to Promotion.

Something like this works well, but chat with your CS team if you need help. And then in the last part of this step, if you want to, you can filter your target metric.

For example, you could use the Event Reason dimension to filter down to specific types of promotions, such as succession or progression, if you wish to do so. But just keep in mind that then your model will only predict if an employee is likely to receive that type of promotion.

Next, select the population that One AI will make predictions on. In this case, this will be which employees we want to predict promotions for. You can use a general headcount end of period metric and filter down in the last part of the step like I'm doing here, or you can select a specific headcount metric like headcount managers or headcount California or whatever for your population metric in order to get to your group of interest.

Then choose a unique identifier for each instance in the model. The unique identifier must come from the same table as the population metric.

When using a headcount metric, person ID, employee ID, worker ID, or any other unique ID from the employee table typically works well. One AI will verify the ID's uniqueness for the model model once selected.

Then you will select the population date, which anchors the query with the predict frame and the train test dataset offset from it. I'm selecting 'Today', which means that this model will use history going back starting from today and will predict a year in the future starting from today as well.

And in the last part, you can add filters to refine your headcount population.

Instead of running the model on the entire organization, you can segment it by factors like department, country, or job level since drivers tend to vary by group. For example, I'm going to use the location dimension to filter down to the Asia Pacific headcount population. In the next step, select how far into the future you want to predict from the population date. This defaults to one year, but can be adjusted to fit your custom modeling needs.

I'm going to leave it at one, which means that this model is going to predict if these employees will be promoted within the next year. Then choose how much historical data you want to use for training the model. We have found that best performance results from enough training intervals to give us 1-2 years of historical data. So because I selected one year in the previous step, each interval equals one year.

I'm leaving a one year so that the model has one year of historical data to learn from. You can adjust this in future runs to see if it impacts model performance by increasing to two or three years.

In the next step, you have the opportunity to add meaningful labels to the values you are predicting. And while this is optional, it greatly simplifies reading your results summary and storyboards.

In this model, null values or zeros indicate 'Not Promoted' while values greater than zero indicate 'Promoted'.

Click 'Load Target Metric Values' to set the overrides.

You can name the groups however you like, such as "No Promotion" and "Promotion", but just ensure that there are only two unique labels total. Without this step, predictions will be sorted into 1s and 0s, making result summaries and storyboards difficult to interpret for end users who don't know which label corresponds to which value.

In the next step, select core attributes, which are the input variables that form the dataset the model learns from and uses to make predictions. So, what attributes about an employee that might impact if they are predicted to be promoted or not.

Some good ones to include are position, tenure, performance, location, etcetera.

We want to aim for balance, enough attributes for robust predictions, but not so many that the results become difficult to interpret.

If you don't intend to manually select each column into your model dataset, I recommend using a narrow or balanced scope to capture attributes that sit in the employee table and or one join away from the employee table. Sometimes performance survey or compensation data is located in its own table, which is when the balance scope is better.

Then go through all of the tables in the included column section and exclude anything you don't want One AI to use using this X button.

Once you've gone through all the included columns and excluded anything you didn't want, you can go through each of the excluded tables and bring in the additional features of interest using this checkmark button.

In the next step, select the generative attributes you want to use in your prediction.

Use the check box to the left of each attribute to include as an input variable or select all by checking the box on top. This step is optional, but highly recommended because generative attributes often improve model performance and robustness greatly.

In the next step, click ‘Generate Data Statistics’ to verify that all of your previous selections are valid. This helps identify errors that cause the model to error while running.

You'll also receive valuable data exploration information about the train test datasets, such as row counts, standard deviations, and min and max values. As you can see, we received the green light "Success," so we can proceed. If One AI finds a problem, it will display "Action Needed" with details about the errors it has found.

And in the final step, you have the option to download the train/test dataset and/or the predict dataset that were generated by this recipe. This is useful for analyzing the data in external tools or for investigating errors encountered in the verification step just before.

Once you're done, review each step, and when you're happy with your selections, click the save icon in the upper right corner, which will prompt you to return to the machine learning model screen.

Here you can manually configure model settings. At minimum, I recommend generating SHAP values.

Refer to our modules on global settings and advanced configuration if interested in this kind of stuff. And once you're complete with your manual configurations, you can scroll to the bottom and click 'Create'. If you don't click create and navigate somewhere else in one model, you will lose your work from the One AI query builder screen.

And just like that, you can now create a promotion classification model in One AI. The next steps are to run your model, then evaluate your results, refine the model, and then deploy and share its insights with stakeholders.

Now let's talk about some of the insights you can analyze from this type of model in One Model.

Chapter 5

Insights to be Analyzed

Section 5 - Insights to be Analyzed

A promotion model offers a range of valuable insights.

First, you can view the model's predictions at the instance level in the Results Explorer, where each employee is classified as either promoted or not promoted based on their unique identifier.

While these predictions are useful, aggregating them at the group level improves accuracy and strategic insight.

I think the real value of a promotion model comes from understanding the drivers behind these outcomes. Using storyboards, you can analyze groups within the model population to identify the factors influencing promotion likelihood.

You can also compare these drivers to the overall population and explore detailed explanations.

For example, these prediction explanations are filtered to the legal department and show that their higher than average engagement scores contribute to their higher scores as a group for promotion.

You can also create geospatial visualizations for likelihoods, and you can use correlation data from the EDA report to explore how different features interact, helping you uncover patterns and relationships that offer deeper insights into what drives promotions within your organization.

If SHAP is enabled, it provides detailed insights into the factors influencing promotion decisions, highlighting the top drivers for promoted and non- promoted employees, ranked by importance.

As you can see here, longer tenure is the second largest driver of promotion.

You can also segment your model population by groupings like job level, performance rating, or department to analyze promotion likelihoods for specific segments.

For example, here we can see that employees with high performance ratings have a higher promotion likelihood compared to other scores. Any dimension that works on headcount will work on this type of chart.

This model also allows for targeted analysis of individual employees. You can examine an employee's promotion likelihood, review their work history, assess how different features impact their chances, and compare their feature values to the overall average.

And finally, you can generate by-name lists of employees predicted to have high, medium, or low likelihood of being promoted, complete with details like their manager, organizational unit, and job level. This allows for more personalized and informed review of potential promotion candidates.

Remember, while predictions are useful, the real value of the model comes from understanding the factors that drive promotion decisions. It's important to use these insights thoughtfully and not rely solely on automated predictions.

For the model to truly guide promotion decisions, it should be accurate, perform well, and be focused on merit based factors.

Chapter 6

Conclusion & Thanks

We've covered how to build a promotion classification model in One AI diving into its use cases, key considerations, and insights. You now have the knowledge to configure, validate, and analyze promotion models tailored to your organization's needs. By focusing on fairness, transparency, and understanding the drivers behind promotions decisions, you can support informed merit based talent management choices. Happy modeling!