tecton

Atlassian Accelerates Deployment of New ML Models From Months to Days and Improves Model Accuracy

ATLASSIAN

Atlassian is a global software company helping teams around the world unleash their potential. They build tools that help teams collaborate, build, and create together, including Jira Software, the #1 software development tool used by agile teams, and Confluence, a team workspacewhere knowledge and collaboration meet. Atlassian products are used by more than 174,000 customers worldwide.

Atlassian aims to provide exceptional experiences to its vast base of daily users. In support of that objective, the Search and Smarts team applies predictive analytics and Machine Learning to power intelligent experiences across its customer ecosystem.

Before engaging with Tecton, one of the Search and Smarts team's most visible projects was to predict and recommend user mentions in Jira and Confluence. Data scientists spent multiple months building the first features and generating training datasets with promising results. But they soon realized that deploying these features in production would be a challenge, particularly due to the use of streaming data.

To serve their features in production, the team set out to build an internal feature store. They assigned ~3 engineers who successfully built the first version in about one year. The internal feature store addressed the most pressing issue of serving features online, but didn't bridge the data science / engineering gap and would not scale to support all of Atlassian's needs. Teams were still operating in silos, it

Challenges:

The complexity, time, and effort required to build and deploy ML data pipelines were slowing down the delivery of ML applications:

- » Features developed in silos with inconsistent handoff of features between data science and engineering
- » Multiple months required to build production pipelines
- » Lack of training / serving parity reducing accuracy of predictions
- » 2–3 FTEs dedicated to managing the internal feature store for ML

still took months of effort to build production pipelines, and the lack of training / serving parity was reducing the accuracy of predictions.

To address these gaps, Atlassian evaluated the Tecton feature store to manage the complete lifecycle of features in Atlassian's operational ML stack. After a successful Proof of Concept, Tecton was deployed in production to power predictive experiences, such as recommending content fields in Jira and Confluence. By using Tecton as their data platform for ML, Atlassian has accelerated the delivery of new features from a range of 1-3 months to just 1 day, and has increased the prediction accuracy of existing models by 2%. These changes are directly improving over 200,000 customer interactions every day in Jira and Confluence.

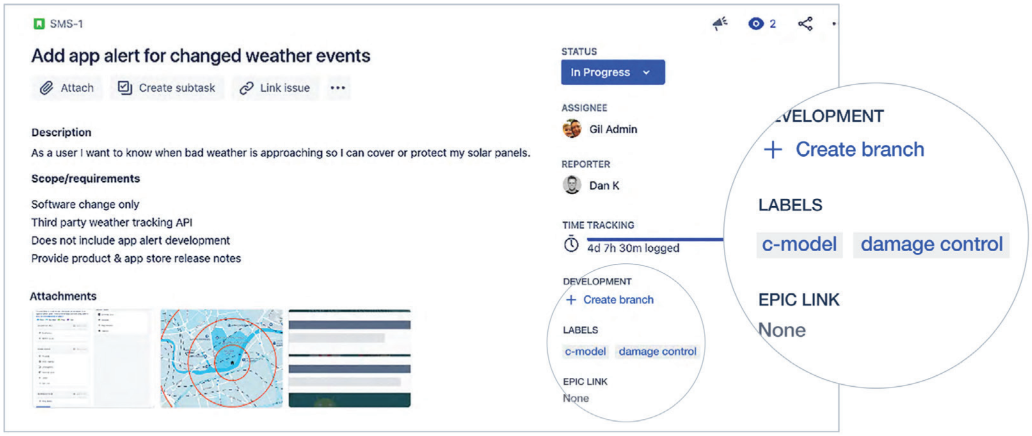
Solution:

Atlassian Implemented the Tecton enterprise feature store to build and deploy ML features as a core component of Atlassian's ML stack. Tecton is now used for multiple production use cases and will support additional use cases in the future:

- » In production: User mention recommendations in Jira and Confluence
- » In development: Content recommendations and in-product searchfor Jira and Confluence
- » On the roadmap: Future ML use cases

ML Use Case Priorities

The Search and Smarts team builds and operates multiple ML applications to improve the customer experience in Atlassian products. For example, the team built an application to predict user mentions, assignees, and labels in Jira and Confluence. User mentions are used to draw someone's attention to a page or comment, while assignees and labels are used to allocate work and categorize content. Instead of simply providing a dropdown list by alphabetical order, models predict and suggest the right content up to 85% of the time. For power users that are assigning 100s of tickets a day, this can meaningfully reduce the time spent on repetitive tasks.



Add app alert for changed weather events

Description

As a user I want to know when bad weather is approaching so I can cover or protect my solar panels.

Scope/requirements

Software change only
Third party weather tracking API
Does not include app alert development
Provide product & app store release notes

STATUS: In Progress

ASSIGNEE: Gil Admin

REPORTER: Dan K

TIME TRACKING: 4d 7h 30m logged

DEVELOPMENT

LABELS: c-model damage control

EPIC LINK: None

Recommended issue labels for every Jira issue

Recommended work allocation and content categorization in Jira

Another high-priority use case is to improve the in-product search experience. Search is used extensively to quickly find relevant pages and to track down issues. It's an essential capability for large teams that may have thousands of pages and issues to search through.

Challenges Getting ML to Production

The Search and Smarts team quickly realized that building and deploying these ML applications at scale would require a new platform for operational ML. Indeed, Atlassian operates at massive scale:

- >174,000 customers
- Billions of events generated every day
- Hundreds of millions of feature-key combinations per model
- >1M individual predictions generated every day

Getting the first models to production at this level of scale was a complex process. Without a feature store in place to manage the lifecycle of ML features, teams were struggling with:

Lack of collaboration tools spanning data science and engineering: The teams didn't have a single repository of features that spanned both development and production environments. Keeping code in sync between data science and engineering required extensive manual coordination through meetings and written documents.

Complexity of incorporating streaming data: Many of Atlassian's features are built with a combination of batch and streaming data to leverage the company's valuable historical information while also capturing the latest events. The use of streaming data introduced unique challenges, such as carefully managing eventtime to prevent data leakage in training datasets and implementing streaming pipelines in production.

Lead times of multiple months to deploy production pipelines: Features created by data scientists were not production-ready.

Results:

Atlassian has improved over 200,000 customer interactions per day with Tecton:

» Accelerated time to build and deploy new features from 1-3 months to 1 day

» Introduced management of 'features as code' and brought DevOps-like practices to feature engineering

» Started building a central hub of production- ready features which will enable feature re-use and collaboration

» Improved the prediction accuracy of existing models by 2%

» Improved the accuracy of online features from 95%–97% accurate to 99.9% accurate

» Freed up 2–3 FTE from maintaining the Atlassian feature store to focus on other priorities

Much of the data scientists' work needed to be reimplemented for production by a separate data engineering team. This included creating and maintaining the infrastructure to process features with streaming data, hardening production pipelines, and adding monitoring to batch pipelines.

Reduced prediction accuracy due to training / serving skew: The pipelines used to serve online and offline data were built with different code bases and different data sources. This introduced online / offline data skew which in turn reduced the accuracy of predictions.

To overcome these challenges, the team set out to build an internal feature store from scratch. Inspired by the Uber Michelangelo platform, Atlassian's original feature store was successfully delivered by a small team of ~3 full-time engineers in one year. The feature store was successful in serving features online, but it was not designed to solve the tooling gap between data science and engineering. Data scientists were still building features in a separate environment from production, and pipelines still needed to be re-implemented in production.

Solution

Tecton was founded by the creators of Uber Michelangelo and provides an enterprise-ready feature store. Atlassian believed that Tecton could address many of its challenges in productionizing ML, while eliminating the need to manage and maintain its own internal feature store.

Atlassian and Tecton ran a Proof Of Concept project. Over the course of a few months, the teams worked together to implement the Tecton enterprise feature store and test the most important use case on Tecton: the models that predict user mentions, assignees, and labels in Jira and Confluence. After the successful POC, Atlassian is now running multiple models in production on Tecton, and has decided to deploy the Tecton feature store as a core component of Atlassian's ML stack for future use cases.

> “
> *As a net result of using the Tecton feature store, we've improved over 200,000 customer interactions every day. This is a monumental improvement for us.”*
>
> **Geoff Sims**
> **Atlassian Data Scientist**

Result

The Tecton enterprise feature store is serving online features for several production models, has improved over 200,000 customer interactions per day, and provides a strong foundation to scale ML at Atlassian. Powered by Tecton, the Search and Smarts team:

Introduced engineering best practices with centrally-managed feature definitions. Data scientists now have a consistent way of defining features as code stored in a Git version-control repository and surfaced to everyone at the company. Data scientists can search and discover existing features, and collaborate more effectively on new feature development. The central feature definitions are used to generate both offline training data and online serving data, providing consistency in the way features are managed across datascience and engineering.

Accelerated the time to build and deploy new features from 1-3 months to 1 day. Data scientists can discover existing features to repurpose across models, and build and deploy new features to production in just 1 day. They can create accurate training datasets with just a few lines of code, relying on Tecton to manage time travel and prevent data leakage. They can serve streaming features to production instantly without depending on data engineering teams to reimplement pipelines. These sorts of timeframes empower data scientists to innovate at a much faster pace.

Improved the prediction accuracy of existing models by 2%. Once migrated to Tecton, prediction accuracy increased by 2% by delivering more accurate data and eliminating training/serving skew. The ease of use of Tecton enabled the teams to build new models that increased prediction accuracy up to 20%.

Improved the accuracy of online features from 95%–97% accurate to 99.9% accurate. Tecton generates both training and serving data from a single pipeline compiled directly from data scientists' feature definitions. By eliminating the need to reimplement pipelines in production, Tecton ensures training/serving parity and can surface more accurate insights.

Reduced the engineering load by 2–3 FTE in the data infrastructure team. Atlassian no longer needs to build and manage its own feature store. This frees up precious engineering time in the data infrastructure team to work on other high-priority projects.

To achieve the company's vision of building smarter, ML-powered user experiences, Atlassian needed an operational ML platform to productionize ML at scale. By using the Tecton enterprise feature store as the data layer of its operational ML stack, Atlassian now has the infrastructure in place to scale ML initiatives and support their longterm product vision. Tecton provides the confidence of a high performing, always-available system that empowers data scientists to build new features and deploy them to production quickly and reliably.