Co-Recommendations Technical Overview

07/17/2025 6:40 pm EDT

At the core of Bluecore's customer experience platform is a **recommendation engine** that customizes each email with products that are most relevant to each individual customer's session, based on **searched categories and terms**, **viewed products**, **products added to cart**, and **purchased products**.

Bluecore's recommendation engine doesn't just algorithmically recommend the products that customers are mostly likely to interact with, but also gives marketers control over the recommendations to ensure a best fit with the desired messaging and campaign.

Because we believe that marketers should have the ability to choose the recommendation algorithm that best fits their needs, Bluecore provides several options from which the marketer can choose.

Recommendation algorithms can also be combined with highly flexible rules, restrictions, and invariants to ensure engaging and coherent messaging.

Additionally, Bluecore constantly monitors the performance of algorithms, and updates them to ensure the best personalized messaging is sent. This article provides an overview of Bluecore's algorithms, as well as the available rules and restrictions for each.

Recommendation algorithms

Before discussing recommendation algorithms, let's establish some terminology first:
- **Input products** are the set of products that get fed into the algorithms to generate recommendations (e.g., products added to cart by a customer in a session).
- **Output products** are the resulting set of products recommended by the algorithm (e.g. those shown in a Cart Abandonment email under a "You May Also Like" section).

Bluecore uses the following recommendation algorithms:
- Attribute-Based Algorithms
- Collaborative-Based Algorithms

Attribute-based algorithms

Attribute-based algorithms understand an input product's characteristics to recommend products that have similar characteristics. For example, if the brand of the input product is Cole Haan and the category is shoes, then the recommended products would be other Cole Haan shoes.

Attribute rank

The Bluecore integration with our client websites collects a series of attributes on each product (e.g. category = shoes, gender = mens and brand = Nike). Given a set of input products, **Attribute Rank** collects all their attributes, and through our propriety search algorithm, returns products that best match those attributes. This ensures that the recommended products best match the input products from an attribute perspective. For example, if a customer was browsing men's Nike basketball shoes, then the recommended products in their personalized Product Abandonment email would be other men's Nike basketball shoes that they had not browsed. Attribute Rank is a good choice for clients with consistent, well structured product catalogs, that want to have the product recommendations in their emails match the attribute structure in their catalogs.

Collaborative-based algorithms

Collaborative-based algorithms use the collective wisdom of customers to identify which products tend to show up in the same session, (i.e., which products tend to be viewed together or which products tend to be bought together). These are Amazon-like algorithms, which may bring to mind the phrase, "Customers who bought this item also bought..."

Best sellers

The simplest collaborative algorithm is **Best Sellers**, which will show the site-wide best sellers. Combining this with rules or other restrictions can make this simple algorithm very powerful. This algorithm is a good choice for merchandising emails that are driven through catalog changes, rather than a user's session on your website. Additionally, it is sometimes used as a secondary recommendation block in conjunction with one of the others.

Co-view

As the name indicates, the **Co-View** algorithm recommends products that tend to be viewed during the same session as the input products– this collaborative-based algorithm was popularized by Amazon's "customers that viewed these items also viewed ...". We are constantly collecting customer browsing sessions on our clients' sites, and we use the vast amount of data to understand collective behavior and then drive these recommendations. This approach is very effective because it can recommend products that do not share any attributes with the input products, but that are still relevant. This algorithm is a good choice for Product Abandonment emails, especially for clients with websites that aren't structured consistently.

Co-cart

Co-Cart follows a similar approach to the Co-View algorithm, except we recommend products that tend to be added to cart with the input products. This algorithm may be a good choice for abandoned cart emails.

Co-purchase

Co-Purchase is similar to the Co-View and Co-Cart algorithms, except we recommend products that tend to be bought with the input products. We typically recommend this algorithm for post-purchase emails.

Custom logic

The recommendation algorithms provide great recommendations; however, sometimes the marketer wants to have more control over what products are recommended. A common example is when business rules prevent certain products or categories from being shown, such as maternity products or products from competing brands.

Additionally, even though we strongly believe that the collective wisdom of the crowd will provide great Co-View, Co-Cart, and Co-Purchase recommendations, some clients may want to ensure the recommendations are more coherent with the input products (e.g., recommended products match the gender of the input products).

Therefore, we provide our clients with a wide array of customizable restrictions that give them the control they need to not only provide the best recommendations but to also ensure the recommendations adhere to their business rules and/or match their campaign vision. Bluecore can layer in the following customizable restrictions:

- **Static Rules**: A static rule must be followed by all products for the campaign. For example, "only show products where the color attribute contains pink" for a one time send going out on Valentine's day.
- **Dynamic Rules**: A dynamic rule gleans information from the input products before it makes its

recommendations, and must be followed by all products for the campaign. This allows the marketer to layer in a level of cohesiveness on an email-to-email basis. For example, if someone looked at 6 products, 4 of which were shirts, we could ensure the recommended products are all shirts.

- **Unique Attribute**: A rule that requires all products in an email to have unique values for a specified attribute (e.g., brand), which allows the marketer to show a wider variety of products.
- **Sorting**: A rule that sorts by specific attributes. The two most commonly used attributes are purchase counts over the last 30 days (highest to lowest) and created date (most recent to least recent). Sorting allows the marketer to include "best selling" or "new arrivals" aspects in their recommendations.

When combined, these rules can take an email to the next level of personalization. For example, we can leverage all four types of custom logic: a **static rule** that requires the products to be pink, a **unique attribute** of brand, **sorting** by purchase counts, and a **dynamic rule** that requires the products must be shirts. The email would then show the best-selling pink shirts (with unique brands) on the site.