

# When Do We Really Need Recommendations?

Tobias Schnabel

## Abstract

Personalized recommendations are a pervasive feature in many productivity applications. They suggest actions to be taken, content to be added, people to be contacted, or files to be opened. In this position paper we argue that far too often the question of “Should we add recommendations?” is foregone in favor of “Can we add recommendations?”, degrading long-term user experience by increasing the complexity of digital ecosystems or software instead of taking away from it. We first talk about a number of factors fueling this development, list a number of consequences, and then end with a discussion of open questions and potential metrics.

## Introduction

Many of the new features of productivity software are powered by recommender systems (RSs). For example, in just Outlook alone, RSs predict which content is relevant for upcoming meetings, what people should be the receivers of an email, when focus time should be scheduled, how sentences should be completed, what short replies should be sent, which emails should be followed up on, or what files should be attached to an email.

### **What has motivated this increased push of recommendation-powered features into productivity platforms and software?**

Focusing on users first, recommendations are often motivated as a solution to the problem of *information overload* as content keeps growing both inside and outside of companies. Second, recommendations, in particular the ones on which actions to take, are thought as a way to improve efficiency by offering shortcuts. On the business side, we argue that an important and commonly overlooked factor, especially in productivity software is (perceived) innovation. Recommendations often do not directly fuel sales as it is the case on e-commerce platforms, but rather have a subtle effect on customers’ perception of a company and their products. In a competing marketplace, there is always pressure to either add new features that distinguish a product from the competitors or add a similar one in order to narrow the gap.

### **What are possible negative consequences of this trend and how can we mitigate these?**

- 1. Blindness to simpler solutions.** Knowing that recommendations are the answer can cause blindness to alternate solutions (e.g., UI change to support faceted search).
- 2. Recommendations overload.** The problem of information overload is transformed into one where there are too many recommendations or RS-powered features to choose from.
- 3. Over-reliance on short-term engagement.** Because of the fast development cycles of RS-powered features, success is often measured through short-term engagement with the recommendations themselves. This is myopic not only because it is blind to negative long-term effects (e.g., loss of trust) but also because it ignores the fact that user attention is bounded and often merely redistributed.

### **How can we mitigate these?**

A very important step is to re-balance and verify incentives and goals. If decreasing information overload is a goal, then this should be tested through user studies measuring cognitive load. Likewise, if the goal is to offer shortcut, we should test whether users become more efficient in user studies or in online experiments. To counteract the innovation pressure and mitigate the three listed consequences from above, a new RS-powered feature needs to demonstrate that it (i) produces better outcomes than any reasonable UI-based change (ii) improves user outcomes in lab studies, (iii) it improves the *entire* product in the *long-term*.

As part of this, many open questions need to be answered, e.g., “What are the right type of system-wide long-term metrics to consider?”, “How can we detect negative long-term outcomes from log data?”, or “Can recommender systems themselves mitigate some of the negative long-term effects?”.