INSIGHTS · TOOLS AND THE FUTURE

# Collaborative Filtering Approaches to Personalized Insight Delivery

The recommendation methods that suggest films and books can also route research findings to the managers most likely to use them. This article explains how collaborative filtering works, where it helps an insights function, and where it misleads.

ModQu research article · 1,826 words · 18 references (APA 7th edition)

Most organisations now produce more research than any one person can read. Survey dashboards, tracker updates, segment profiles, competitor scans and qualitative summaries pile up in shared drives and insight platforms. Reviews of information overload in business show that when the volume of material exceeds a person's capacity to process it, decision quality falls rather than rises (Edmunds & Morris, 2000; Eppler & Mengis, 2004). The problem for an insights team is therefore not only to generate findings, but to get the right finding to the right person at the right time.

Consumer platforms faced the same problem decades ago, with films, books and news articles. Their main answer was **collaborative filtering**: recommending items to a person on the basis of what similar people found useful. This article explains how collaborative filtering works, how it can be adapted to deliver research insights inside an organisation, and what its known weaknesses mean for anyone who tries.

## Where collaborative filtering came from

The term was introduced by Goldberg et al. (1992) for Tapestry, an experimental email system at Xerox PARC. Users could annotate messages, and other users could filter their mail using those annotations, for example by asking for documents that a trusted colleague had marked as useful. The filtering was "collaborative" because people helped one another find relevant material simply by recording their reactions.

GroupLens automated the idea for Usenet news (Resnick et al., 1994). Readers rated articles, the system found readers whose past ratings resembled each other, and it predicted how much a reader would like an unread article from the ratings of those similar readers. Resnick and Varian (1997) later used the term **recommender systems** for this whole family of tools, which by then included recommendation services for music, films and books.

## How it works

Collaborative filtering starts with a matrix. Rows are users, columns are items, and each cell holds a rating, a click, a purchase or some other signal of value. Most cells are empty, because each person has used only a small fraction of the items. The task is to predict the empty cells and recommend the items with the highest predictions.

## User-based methods

The original approach finds users who resemble the target user and averages their opinions. Similarity is usually measured with a correlation or cosine measure computed on the items both users have rated. The prediction for an unrated item is the target user's average rating plus a weighted average of how far each similar user's rating of that item lies above or below their own average (Resnick et al., 1994; Herlocker et al., 2004).

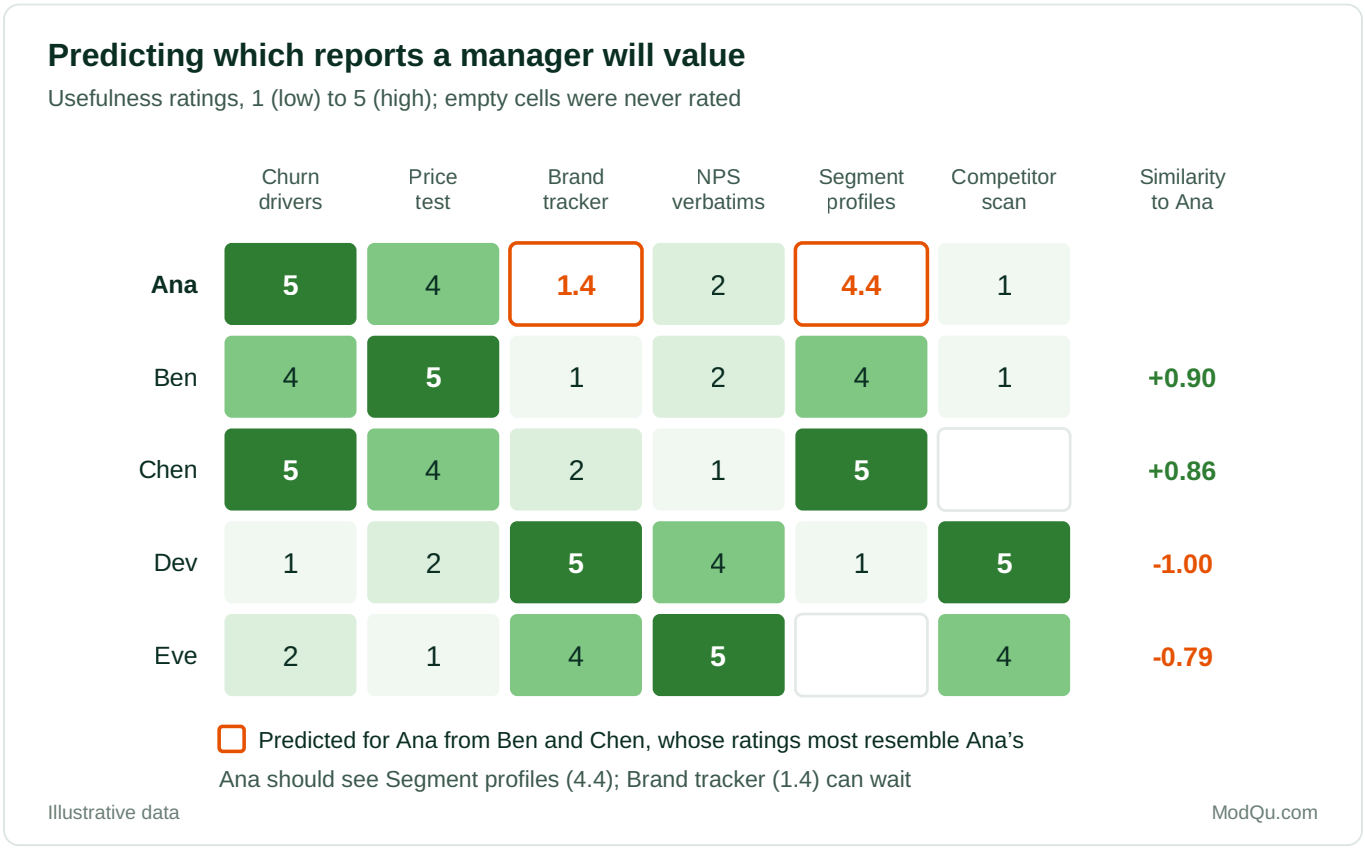



Figure 1. A user-based collaborative filtering example. Ben's and Chen's ratings move with Ana's, so their opinions of the two reports Ana has not opened are used to predict Ana's. Illustrative data; chart by ModQu.

Figure 1 shows a small hypothetical example. Five managers have rated six research outputs for usefulness. Ana has not yet opened the brand tracker or the segment profiles. Ben and Chen rate reports much as Ana does (correlations of +0.90 and +0.86 on shared items), while Dev and Eve rate them in the opposite way. Using only the positively correlated neighbours, Ana's predicted rating is 4.4 for the segment profiles and 1.4 for the brand tracker. An insight platform would therefore put the segment profiles at the top of Ana's list.

### Item-based methods

User-based methods become slow when there are many users, and user tastes change. Sarwar et al. (2001) proposed computing similarity between items instead: two reports are similar if the same people tend to value both. Amazon described a production system built on the same item-to-item logic, which allowed recommendations to be computed quickly for very large catalogues (Linden et al., 2003). In an insights setting, item-based filtering answers a familiar request: "people who used this churn analysis also found these three reports useful."

### Latent factor models

The most accurate approaches in the Netflix Prize competition represented both users and items as short vectors of hidden factors, learned so that the product of a user's vector and an item's vector approximates the observed ratings (Koren et al., 2009). In research delivery, such factors might correspond to themes such as pricing, customer experience or brand, although the model does not name them and they need not be interpretable.

### Implicit feedback

Managers rarely rate reports. They open them, read them for a time, download them, forward them or cite them in a decision paper. Hu et al. (2008) adapted matrix factorisation to such **implicit feedback**, treating each action as evidence of preference held with a stated level of confidence rather than as an explicit score. For an internal insight platform this is the realistic case: usage logs are abundant, ratings are scarce.

## Applying it to insight delivery

Recommender systems are normally used to sell products. Inside an organisation the goal is different: to raise the chance that a relevant finding reaches a decision-maker before a decision is made. That changes how the methods should be applied.

**Define what counts as value.** A click is a weak signal; a report referenced in a business case is a strong one. Teams should decide which actions count as evidence that an insight was used, and weight them accordingly.

**Combine content and collaboration.** Pure collaborative filtering knows nothing about what a report says. **Hybrid recommenders** combine usage patterns with item content, such as the topic, market and method of each report (Burke, 2002; Adomavicius & Tuzhilin, 2005). Tagging every research output with consistent metadata makes this possible and also improves ordinary search.

**Plan for cold starts.** A newly published study has no usage history, so a purely collaborative system will never recommend it. Schein et al. (2002) called this the cold-start problem and showed how content attributes can stand in for missing ratings. New insights, which are often the most valuable, should therefore be pushed through content rules or editorial selection until usage builds up.

**Respect the decision calendar.** A recommendation is useful only if it arrives before the decision it can inform. Linking recommendations to planning cycles, product reviews or campaign launches matters more than small gains in prediction accuracy.

## A staged approach for an insights team

Few insights teams need a full recommender engine on the first day. A staged approach keeps the effort in proportion to the data.

1. **Catalogue the research.** Give every report, dashboard and summary a consistent record: topic, market, customer segment, method, date and owner. This is content data, and it supports search and simple rules even before any usage is recorded.
2. **Log meaningful use.** Record opens, time spent, downloads, shares and, where possible, links between a report and a later decision document. Agree with staff which of these signals will be stored and for how long.
3. **Start with item-to-item suggestions.** Once a few hundred uses are logged, show "often used together" lists on each report page. Item similarities are more stable than user similarities when the user base is small, and they are easy to explain.
4. **Add personal digests.** When usage history is deep enough, send each manager a short periodic digest that mixes personalised predictions with a few editorially chosen findings and every newly published study in that manager's area.
5. **Review and adjust.** Every quarter, check which recommended insights were used in decisions, which important studies were never recommended, and whether some groups of managers see a narrower range of findings than others.

Each stage produces something useful on its own, and each stage generates the data the next one needs. Teams that stop at stage three still gain a better-organised library and a clearer view of what research is actually used, which is valuable evidence for planning future studies.

## Measuring whether it works

Herlocker et al. (2004) reviewed how collaborative filtering systems are evaluated and showed that conclusions depend heavily on the metric chosen. Prediction error measures how close predicted ratings are to actual ratings; ranking and coverage measures ask whether useful items appear near the top of the list and whether the system can recommend most of the catalogue. McNee et al. (2006) argued that accuracy alone is not enough, because a list of highly accurate but obvious recommendations adds little.

Later work formalised these "beyond-accuracy" goals as diversity, novelty, serendipity and coverage (Kaminskas & Bridge, 2016). For an insights function they matter a great deal. A finding that a manager would not have searched for, but that changes a plan, is worth more than ten reports the manager would have found anyway. Commercial experience points the same way: Netflix described the value of its recommender system in terms of member engagement and retention, tested through online experiments, rather than rating accuracy alone (Gomez-Uribe & Hunt, 2015).

A practical evaluation plan for an internal system might compare a group of managers who receive personalised recommendations with a group who see a standard list, and track whether recommended insights are opened, cited in decisions and rated as useful afterwards.

## Risks and limits

**Filter bubbles.** Pariser (2011) warned that personalisation can narrow what people see to what they already agree with. Inside a company, the equivalent risk is that a sales director sees only sales research and never the customer-experience finding that explains falling renewals. Deliberately mixing in diverse or contrary findings reduces this risk.

**Popularity bias.** Collaborative filtering favours items that many people have used. Widely read trackers will crowd out niche but important studies unless the system corrects for popularity.

**Small populations.** Consumer recommender systems work with millions of users. An insights team may serve fifty managers. With so few users, similarity estimates are unstable, and simple content rules or curated lists can outperform statistical models. The methods in the *Recommender Systems Handbook* (Ricci et al., 2015) should be scaled to the data actually available.

**Privacy and trust.** Usage logs reveal what each manager reads. Staff should know what is recorded and how it is used, and logs should not be repurposed to judge individuals.

**Feedback loops.** If a system only shows what it predicts people want, it learns only from what it showed. Periodic random or editorial selections give the system evidence about items it would otherwise ignore.

## Key takeaways

Collaborative filtering predicts what a person will value from the recorded behaviour of people like them. Its user-based, item-based and latent factor forms are well established, and they can help an insights team route findings to the managers most likely to use them. To work inside an organisation, the methods need clear definitions of value, content metadata for new studies, timing linked to decisions and deliberate safeguards against narrowing what people see. The aim is not more clicks on reports, but more decisions informed by the evidence the organisation has already paid for.

## References

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering, 17*(6), 734–749. https://doi.org/10.1109/TKDE.2005.99

Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction, 12*(4), 331–370. https://doi.org/10.1023/A:1021240730564

Edmunds, A., & Morris, A. (2000). The problem of information overload in business organisations: A review of the literature. *International Journal of Information Management, 20*(1), 17–28. https://doi.org/10.1016/S0268-4012(99)00051-1

Eppler, M. J., & Mengis, J. (2004). The concept of information overload: A review of literature from organization science, accounting, marketing, MIS, and related disciplines. *The Information Society, 20*(5), 325–344. https://doi.org/10.1080/01972240490507974

Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM, 35*(12), 61–70. https://doi.org/10.1145/138859.138867

Gomez-Uribe, C. A., & Hunt, N. (2015). The Netflix recommender system: Algorithms, business value, and innovation. *ACM Transactions on Management Information Systems, 6*(4), Article 13. https://doi.org/10.1145/2843948

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems, 22*(1), 5–53. https://doi.org/10.1145/963770.963772

Hu, Y., Koren, Y., & Volinsky, C. (2008). Collaborative filtering for implicit feedback datasets. In *Proceedings of the Eighth IEEE International Conference on Data Mining* (pp. 263–272). IEEE. https://doi.org/10.1109/ICDM.2008.22

Kaminskas, M., & Bridge, D. (2016). Diversity, serendipity, novelty, and coverage: A survey and empirical analysis of beyond-accuracy objectives in recommender systems. *ACM Transactions on Interactive Intelligent Systems, 7*(1), Article 2. https://doi.org/10.1145/2926720

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer, 42*(8), 30–37. https://doi.org/10.1109/MC.2009.263

Linden, G., Smith, B., & York, J. (2003). Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing, 7*(1), 76–80. https://doi.org/10.1109/MIC.2003.1167344

McNee, S. M., Riedl, J., & Konstan, J. A. (2006). Being accurate is not enough: How accuracy metrics have hurt recommender systems. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems* (pp. 1097–1101). ACM. https://doi.org/10.1145/1125451.1125659

Pariser, E. (2011). *The filter bubble: What the Internet is hiding from you*. Penguin Press.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994). GroupLens: An open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work* (pp. 175–186). ACM. https://doi.org/10.1145/192844.192905

Resnick, P., & Varian, H. R. (1997). Recommender systems. *Communications of the ACM, 40*(3), 56–58. https://doi.org/10.1145/245108.245121

Ricci, F., Rokach, L., & Shapira, B. (Eds.). (2015). *Recommender systems handbook* (2nd ed.). Springer. https://doi.org/10.1007/978-1-4899-7637-6

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web* (pp. 285–295). ACM. https://doi.org/10.1145/371920.372071

Schein, A. I., Popescul, A., Ungar, L. H., & Pennock, D. M. (2002). Methods and metrics for cold-start recommendations. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 253–260). ACM. https://doi.org/10.1145/564376.564421