

**Е. В. Савлук**  
(ГрГУ им. Я. Купалы, Гродно)

## **РАЗРАБОТКА РЕКОМЕНДАТЕЛЬНОГО СЕРВИСА**

Рекомендательный сервис – это программа, которая пытается предсказать, какие объекты будут интересны пользователю, имея определенную информацию о его предпочтениях. Такая система – частный случай экспертной системы, экспертную работу в которой выполняют сами пользователи сервиса.

Одним из наиболее эффективных и популярных способов построения прогнозов в рекомендательных системах является коллаборативная фильтрация. Алгоритмы коллаборативной фильтрации для прогнозирования предпочтений пользователя используют уже известные предпочтения группы пользователей. Прогнозы составляются индивидуально для каждого пользователя, хотя используемая информация собрана от многих участников. Тем самым коллаборативная фильтрация отличается от более простого подхода, дающего усреднённую оценку для каждого объекта интереса.

Был спроектирован рекомендательный сервис, позволяющий пользователю получить рекомендацию книг к прочтению. Для каждого пользователя книги делятся на три категории: уже прочитанные, те, которыми он заинтересовался и собрался прочесть, и рекомендованные к прочтению. После регистрации пользователь заполняет свою «полку» прочитанными книгами, выставив каждой из них оценку по шкале от одного до пяти. Таким образом, прочитанные книги дают информацию о предпочтениях пользователя и позволяют сервису составить рекомендацию.

Для прогнозирования предпочтений использован подход коллаборативной фильтрации основанный на соседстве. Так, для прогнозируемого пользователя подбирается подгруппа пользователей схожих с ним в оценке прочитанных книг, а затем комбинация весов и оценок этой подгруппы используется для прогноза оценок пользователя для неоцененных им книг. На основе этого прогноза и составляется рекомендация пользователю.

Платформой разработки выбрана Java. Проект реализован как web-приложение. Клиентская часть выполнена с помощью фреймворка PrimeFaces, который позволяет быстро реализовывать кроссплатформенный web-интерфейс. Серверная часть основана на Spring для общего каркаса приложения и Hibernate для ORM. В качестве персистентного хранилища данных использована реляционная база данных MySQL.