Recommender Systems

Course Syllabus for CS 597 (Special Topics: Recommender Systems), Spring 2017

Overview

This course covers the basic concepts of recommender systems, including personalization algorithms, evaluation tools, and user experiences. We will discuss how recommender systems are deployed in e-commerce sites, social networks, and many other online systems, with many readings from current research in the field.

Course Logistics

Course Title	CS 597
Credits	3
Schedule	Tu/Th 4:30-5:45 PM in CCP 352
Prerequisites	CS 321 (Data Structures) or equivalent
Readings	*Practical Recommender Systems* (textbook)
	Recommender Systems: An Introduction (optional textbook)
	Collaborative Filtering Recommender Systems
	Additional research papers and online articles
Software	LensKit, Java, data analysis (R, Python, or Excel)

Instructor

Michael Ekstrand

Office	CCP 255
E-mail	michaelekstrand@boisestate.edu
Phone	(208) 426-5761
Office Hours	M 11 AM-12 PM, Tu/Thu 3:30-4:30 PM

Resources and Readings

Textbook

Our primary (and required) textbook is *Practical Recommender Systems* by Kim Falk Jørgensen (Manning, 2017).

You may also find the video lectures in the *Recommender Systems* specialization on Coursera a useful addition to your study. The lectures are free; payment is only required if you wish to take the Coursera assignments and get a certificate.

Supplemental Books

The following optional textbook may be useful if you wish to dive deeper into some of the topics we cover in a more traditional textbook:

> *Recommender Systems: An Introduction* by Jannach, Zanker, Felfernig, and Friedrich (Cambridge, ISBN 978-0-52-149336-9)

You will likely find this survey article useful to read:

> *Collaborative Filtering Recommender Systems* by Ekstrand, Riedl, and Konstan (now publishers; Foundations and Trends in Human-Computer Interaction 4(2)). Available free from my web site.

Finally, the *Recommender Systems Handbook* is a good resource for overviews of many topics in recommender systems.

Research Readings

Throughout the course, we will be reading and discussing a number of research papers. The currently-planned list of papers is in the appendix of this syllabus; links to each paper will be provided in Blackboard.

Software Resources

This course contains several programming assignments which will be in Java using the LensKit framework. If you do not yet know Java, there are several resources available to help you learn it; I will provide links to some on Blackboard.

To study the principles of good Java application design and coding practices and better understand the LensKit software, I recommend the following book:

> Joshua Bloch. 2008. *Effective Java* 2nd ed., Upper Saddle River, NJ: Addison-Wesley. ISBN 978-0-201-31005-4.

In addition, you will likely need to consult documentation for LensKit and other Java libraries; further links to these resources will be provided in Blackboard.

The current LensKit documentation can be found at http://mooc.lenskit.org.

COURSE STRUCTURE

I have organized the class material into *modules*, each on a different topic.

You are responsible for reading the textbook and/or watching the online videos. We do not have time to discuss everything that you need to know in class; our time there is better spent diving deeper into the topics.

Course Components

The work in this course falls into several categories:

- **Reading reports** ask you to briefly summarize a research paper from the readings.
- **Assignments** to give you experience building, testing, and analyzing recommender systems.
- A **research project** to understand, communicate, and extend recent recommender systems research.
- **Midterm** and **final** exams.

Your final grade will be computed from these components as follows:

Category	*%*
Reading Reports	5
Assignments	40
Exams	30
Research Project	25

The standard 70/80/90 scale determines the minimum grade you will receive (that is, if you have 80 total course points, you will receive at least a B-).

Readings and Reports

Each week has an assigned research reading. These readings will also form the basis of the Reading Reports. For each week's reading, write 2–3 paragraphs briefly addressing the following:

- What question is the paper trying to answer, or what problem is it trying to solve?
- What is the key idea of the solution or experimental method?
- What are the main lessons from the paper?

In addition to the assigned course readings, you must find and read some additional research papers. You must submit a total of **12 reading reports**: 10 from the assigned readings and 2 from additional research papers.

Do not neglect the reading reports. The research project will be *much* easier if you are regular in doing the readings and their reports.

I also encourage you to discuss the readings with your fellow students in advance of class; you may even do this before submitting the reading reports. The reading report itself must be your own work, but discussion is fine.

Assignments

There will be several assignments throughout the semester. Many of these assignments will require you to implement and/or test one or more recommender algorithms using the LensKit toolkit.

Some assignments will involve analyzing or designing recommender applications.

Each assignment is due at **midnight on Tuesday** of the week in which it is due.

You may work in groups of up to 2 on the assignments. If you work with a group, only submit the assignment under **one group member**, and list all members of your group in your assignment submission.

Research Project

The final component of this class is the research project. There are two options:

- A survey and research proposal, surveying and summarizing at least 4 research papers and proposing a new experiment to extend their results or answer new questions they raise.
- A replication study, where you attempt to reproduce a recent recommender systems algorithm or experiment paper.

I encourage you to work with a partner on this project. Each student or group will present on their research project at the end of the semester. More details will be available in the research project description.

COURSE POLICIES

Attendance

You should attend all class sessions if possible. If you need to be absent for some reason, such as conference travel or illness, please let me know as soon as you can.

Late Work

For the **assignments**, you may submit assignments late with a 10% penalty for each day late.

For the **research project deliverables**, each deliverable must be submitted on time. Deliverables will be accepted up to 24 hours late with a 25% grade penalty.

Late reading reports are not accepted, but only the required number of reading reports will affect your grade.

The mid-term and final exams will be at the published times. Makeup exams will only be given in exceptional circumstances.

Cheating and Academic Integrity

As both a programmer and a student, you are expected to do your own work, attribute sources, and respect the legal and moral rights of others with respect to their work; as a student, you are also required to abide by the university policies regarding academic integrity. While I aim to allow you to make reasonable use of resources, cheating (including copying code, using unauthorized resources during tests, etc.) will not be tolerated. If you are found to be cheating, the penalty may range from an F on the assignment to an F on the course, and will also be reported to the university.

External Resources

You may consult external resources such as other books and web sites for understanding how to solve assignments or portions of the project. In your assignment solution, list all external resources you used; if they are available online, provide the URL. You do not need to cite the textbook, or the official documentation for the software we are using (Java, LensKit, and the libraries used by LensKit).

Besides the course forum, you may ask questions related to the course material and concepts required to complete the work on publicly accessible discussion forums such as Stack Overflow, newsgroups, or publicly-archived mailing lists. To qualify as publicly-accessible, a site must provide access to complete discussions without requiring payment or registration. Provide URLs to the forum discussion on the relevant web site or archive (Google Groups works well for newsgroup archives) with your project deliverable submission. When you ask a question for one of the assignments, mention that it is for a course project and that your instructor permits you to make reasonable use of discussion forums. You may also freely use the LensKit Gitter chat room.

A good question will ask about how to go about a particular sub-portion of the problem, how something works, why something you are trying doesn't work, or other specific difficulties. Do not ask "how do I solve <the problem description>?", or similarly direct translations of the project requirements, or for specific code. Questions should be written to fill in a gap in your understanding that will then enable you to continue your work, not to get a solution to the assignment.

Conduct

I expect you to behave in a civil, respectful manner in all class interactions, both in official meetings such as lectures and out-of-classroom activities such as project group meetings and study sessions, and to contribute to a constructive learning environment.

The Recurse Center Social Rules are a good source of guidance on how to maintain a constructive and educational environment.

If you experience or witness harassment of any form, please let me know.

Disability Accommodations

If you need particular accommodations to be able to fully participate in this course, please talk with me as soon as possible. I may ask that you provide documentation from the Office of Disability Services, so if you have such documentation please bring it.

SCHEDULE

Following is an approximate schedule. I may adjust it as we progress through the semester. **Bold** items are key dates for the project and exams.

Week	Date	Topic	Chapter	Assignment
1	1/9	Introducing Recommenders	1	
2	1/16	User Input in Recommendation	2,4	
3	1/23	Non-Personalized Recommendations	3,5	A1
4	1/30	Content-Based Recommendations	7,9	A2
5	2/6			
6	2/13	Nearest-Neighbor Collaborative Filtering	8	A3
7	2/20			
8	2/27	Evaluating Recommenders	13	A4
	3/6	**Midterm** (Thursday)		*Research Proposal*
9	3/13	Presenting and Explaining Recommendations	6	A5
10	3/20	*Spring Break*		
11	3/27	Matrix Factorization	10	A6
12	4/3	Machine Learning		
13	4/10	Hybrid Recommenders	11	A7
14	4/17	Learning to Rank	12	*Draft Report*
15	4/24	**Research Presentations**	15	A8
F	5/1	**Final Exam** (Thu. May 4 3-5 PM)		*Final Report*

Assignments (A*n*) are due **on Tuesday at midnight** the week they are listed.

APPENDIX A: RESEARCH READINGS

Below is the tentative list of class readings. You may need to be on campus to access full papers from the links below; links that work off-campus will be available on Blackboard. While I don't expect it to, this list may change as we move through the semester; if I change a reading, I will notify you at least the week before it is due.

Week 1

Hill, Will, Larry Stead, Mark Rosenstein, and George Furnas. 1995. "Recommending and Evaluating Choices in a Virtual Community of Use." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 194–201. CHI '95. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co. doi:10.1145/223904.223929.

Week 2

Amatriain, Xavier, Josep M. Pujol, and Nuria Oliver. 2009. "I Like It... I Like It Not: Evaluating User Ratings Noise in Recommender Systems." In *User Modeling, Adaptation, and Personalization*, edited by Geert-Jan Houben, Gord McCalla, Fabio Pianesi, and Massimo Zancanaro, 247–58. Lecture Notes in Computer Science 5535. Springer Berlin Heidelberg. doi:10.1007/978-3-642-02247-0_24.

The following are optional, supplementary papers for Week 2:

Cosley, Dan, Shyong K. Lam, Istvan Albert, Joseph A. Konstan, and John Riedl. 2003. "Is Seeing Believing?: How Recommender System Interfaces Affect Users' Opinions." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 585–92. CHI '03. New York, NY, USA: ACM. doi:10.1145/642611.642713.

Kluver, Daniel, Tien T. Nguyen, Michael Ekstrand, Shilad Sen, and John Riedl. 2012. "How Many Bits Per Rating?" In *Proceedings of the Sixth ACM Conference on Recommender Systems*, 99–106. RecSys '12. New York, NY, USA: ACM. doi:10.1145/2365952.2365974.

Week 3

Senecal, Sylvain, and Jacques Nantel. "The Influence of Online Product Recommendations on Consumers' Online Choices." *Journal of Retailing* 80, no. 2 (2004): 159–69. doi:10.1016/j.jretai.2004.04.001.

Week 4

Phelan, Owen, Kevin McCarthy, Mike Bennett, and Barry Smyth. 2011. "Terms of a Feather: Content-Based News Recommendation and Discovery Using Twitter." In *Advances in Information Retrieval*, edited by Paul Clough, Colum Foley, Cathal Gurrin, Gareth J. F. Jones, Wessel Kraaij, Hyowon Lee, and Vanessa Mudoch, 448–59. Lecture Notes in Computer Science 6611. Springer Berlin Heidelberg. doi:10.1007/978-3-642-20161-5_44.

Week 5

Pera, Maria Soledad, and Yiu-Kai Ng. 2014. "Automating Readers' Advisory to Make Book Recommendations for K-12 Readers." In *Proceedings of the 8th ACM Conference on Recommender Systems*, 9–16. RecSys '14. New York, NY, USA: ACM. doi:10.1145/2645710.2645721.

Week 6

Sarwar, Badrul, George Karypis, Joseph Konstan, and John Reidl. 2001. "Item-Based Collaborative Filtering Recommendation Algorithms." In *ACM WWW '01*, 285–95. doi:10.1145/371920.372071.

Week 7

Golbeck, Jennifer. 2006. "Generating Predictive Movie Recommendations from Trust in Social Networks." In *Trust Management*, edited by Ketil Stølen, William H. Winsborough, Fabio Martinelli, and Fabio Massacci, 93–104. Lecture Notes in Computer Science 3986. Springer Berlin Heidelberg. doi:10.1007/11755593_8.

Week 8

Said, Alan, Ben Fields, Brijnesh J. Jain, and Sahin Albayrak. 2013. "User-Centric Evaluation of a K-Furthest Neighbor Collaborative Filtering Recommender Algorithm." In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work*, 1399–1408. doi:10.1145/2441776.2441933.

Week 9

Rossetti, Marco, Fabio Stella, and Markus Zanker. 2016 "Contrasting Offline and Online Results When Evaluating Recommendation Algorithms." In *Proceedings of the 10th ACM Conference on Recommender Systems*, 31–34. doi:10.1145/2959100.2959176.

Week 10

Vig, Jesse, Shilad Sen, and John Riedl. 2009 "Tagsplanations: Explaining Recommendations Using Tags." In *Proceedings of the 13th International Conference on Intelligent User Interfaces - IUI '09*, 47. doi:10.1145/1502650.1502661.

Week 11

Bollen, Dirk, Bart P. Knijnenburg, Martijn C. Willemsen, and Mark Graus. 2010. "Understanding Choice Overload in Recommender Systems." In *Proceedings of the Fourth ACM Conference on Recommender Systems*, 63–70. RecSys '10. New York, NY, USA: ACM. doi:10.1145/1864708.1864724.

Week 12

Christakopoulou, Evangelia, and George Karypis. 2016. "Local Item-Item Models For Top-N Recommendation." In *Proceedings of the 10th ACM Conference on Recommender Systems*, 67–74. doi:10.1145/2959100.2959185.

Week 13

Eugene Agichtein, Eric Brill, and Susan Dumais. 2006. "Improving web search ranking by incorporating user behavior information." In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (SIGIR '06). doi:10.1145/1148170.1148177

Week 14

Rendle, Steffen, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. "BPR: Bayesian Personalized Ranking from Implicit Feedback." In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, 452–461.

Week 15

Hariri, Negar, Bamshad Mobasher, and Robin Burke. 2014. "Context Adaptation in Interactive Recommender Systems." In *Proceedings of the 8th ACM Conference on Recommender Systems*, 41–48. RecSys '14. New York, NY, USA: ACM. doi:10.1145/2645710.2645753.