Recommender Systems || Lecture 9

Summer 2022

Dr. Karthik Mohan

UW, Seattle

July 27, 2022

Logistics

1. Assignment 2 due tonight!

Question → Discord

Diminishing stepsize / lr

$lr_i = \frac{lr_0}{(epoch+1)}$

(speeds up learning)

Gradient step right

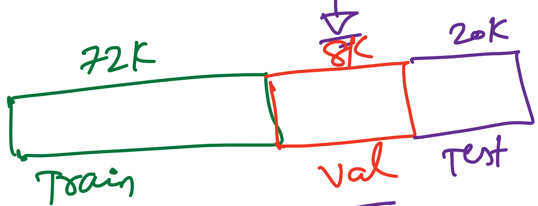
U.grad

V.grad

Good starting pt.

torch.rand( )

Hyper-param search on validation



Reasonable $\begin{cases} U = \text{torch.rand(,)} \\ V = \text{torch.rand(,)} \end{cases}$

$X = U\Sigma V^T$ } SVD

$U_0 = U\Sigma^{1/2}$ $V_0 = \Sigma^{1/2}V^T$ } Starting pt. from SVD!

Logistics

1. Assignment 2 due tonight!
2. Please pick a time slot for your team to discuss project propsoal if you haven't yet!

Logistics

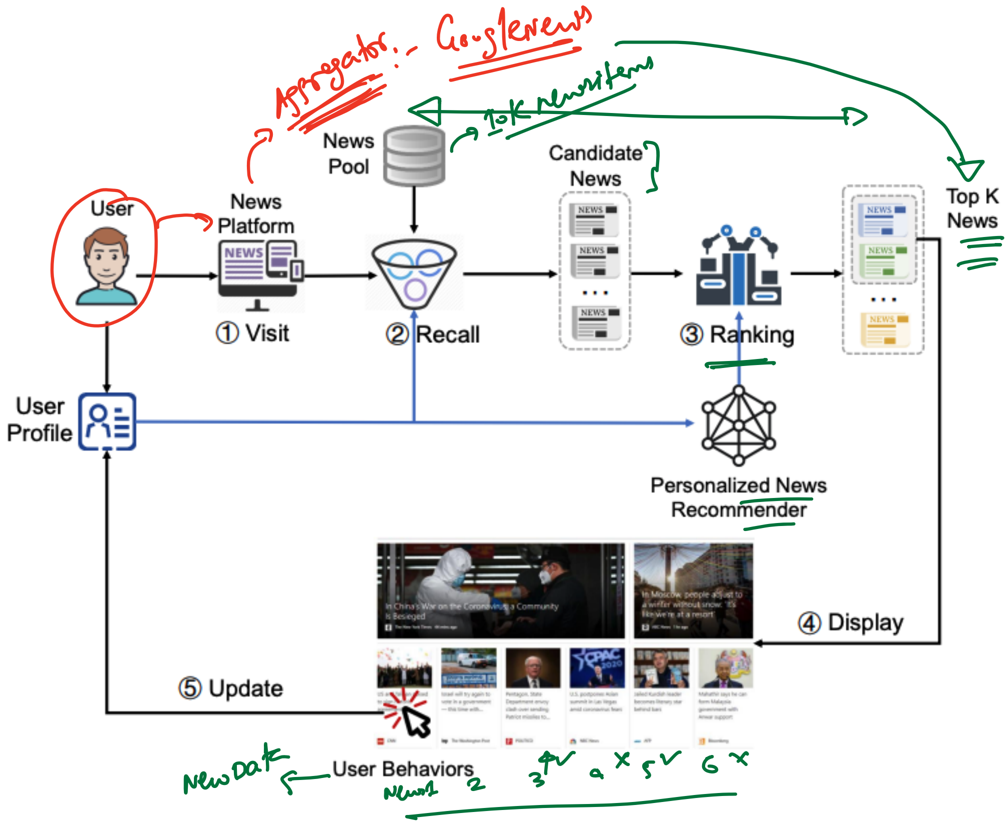
1. Assignment 2 due tonight!
2. Please pick a time slot for your team to discuss project propsoal if you haven't yet!
3. Anything else?!

Today

1. News Recommendations Case Study

↳ Assignment 3

News Recommendations



Reference paper

Challenges for news recommedation systems

Contrast with a product on Amazon (1 year or 2 years)

1. News articles have a short shelf life (maybe a couple days?)

Challenges for news recommedation systems

1. News articles have a short shelf life (maybe a couple days?)
2. So huge cold start problem of new news articles that are waiting to be recommended

Challenges for news recommedation systems

1. News articles have a short shelf life (maybe a couple days?)
2. So huge cold start problem of new news articles that are waiting to be recommended
3. No explicit ratings of news articles by users (like for movies lens data set)

But have clicks

Contrast:- Amazon Reviews (1 star to 5 star)

1) Click Data:- Informative BUT with noise

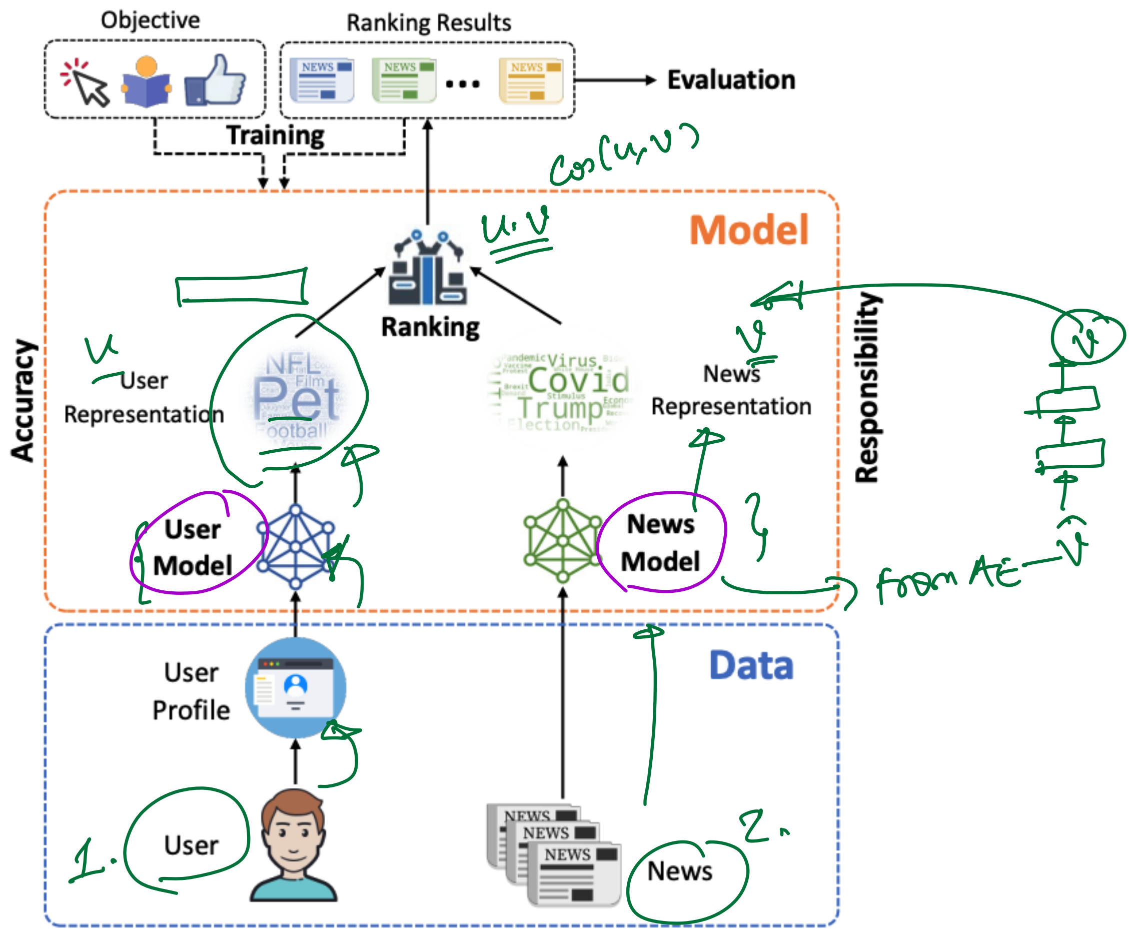
2) Engagement Data:- Time spent on a news article

Breakout #1

Building your news recommender systems

You are a data scientist working on a news recommendation system. You notice that there are news articles that suddenly become "hot" but also seem to taper off in views after a couple of days. You also notice that there are a large number of news articles that don't seem to get many hits and also taper off on views. You are tasked with making these "cold" and "warm" news articles become more "hot" through recomemndations and personalization. However as we discussed, there are challenges to this due to the cold start problem. Brainstorm your approach to building a reasonable news recommender system in this kind of a scenario. What specific machine learning approaches would you take and what could be an architecture for your model?

News Recommendation generic Modeling approach

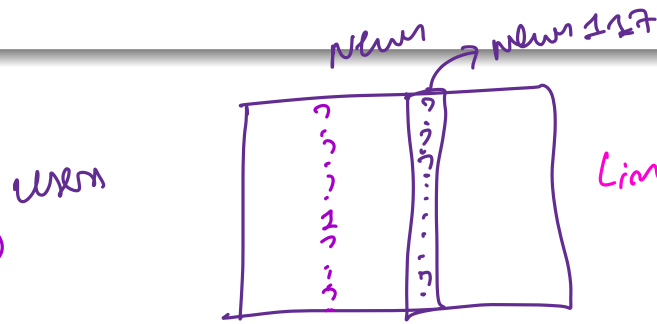


Breakout #2

Pros/Cons of Collaborative Filtering

One obvious method to obtain user and news representations is to use a collaborative filtering approach. We can have a matrix with perhaps 1's and zeros. 1 if a user has viewed a news article and 0 if not. Brainstorm the pros/cons of using this approach for news recommendations. What unique challenges would you encounter? How would you improve upon this approach?

use content info.
(Title, body text, etc)



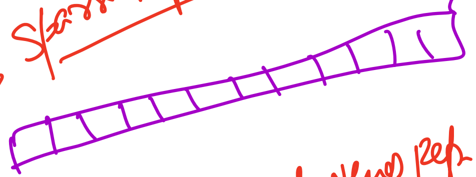
News → id = 117 → v_{117}?

Limited: id → v_{id} (behavior)

id, content(id) → v_{id} (content + behavior)

News Modeling

Sparse Representation



BoW model but weighted by TF-IDF!

TF-IDF/SF-IDF

XF-IDF

↳ (T, C, S)

1) Feature-based News Rep.

Term Frequency

Inverse Document Frequency

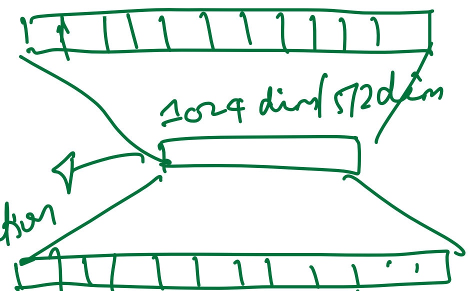
News 1 → TF-IDF (T-Term)

↳ CF-IDF (C-Concept)

↳ SF-IDF (S-Synonym)

2) Deep Learning based News Modeling

(De-noising AE)



Dense News Representation

v

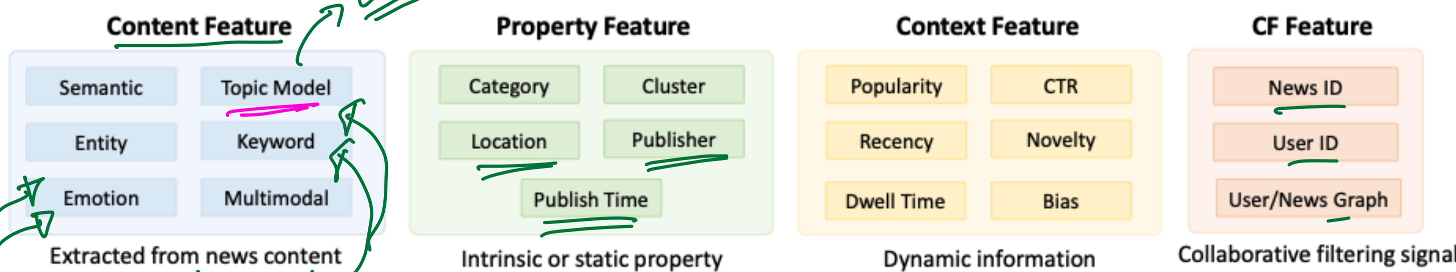
News Recommendation Features and Feature Stores

Cold-Start

unsupervised

LDA - topic models

(user/content)



Title, Body

Reference paper

Content Store

Content Store

News Recommendation Papers by Features used

svd

Features for News Modeling	References
BOW/XF-IDF*	[58][16][140][17][67][68][18][30][11][55][25][99][202][142][163] [236][7][47][14][59][94][95][132][154][147][148][123][201][129]
Entity/Keyword	[58][16][140][17][67][68][18][30][11][109][55][187][75][118][141][108][84] [26][28] [163][236][7][47][244][13][14][15][44][85][95][191][89][195]
Cluster/Category	[104][109][82][28][41][25][122][171][28][182][48][112][111][193][106][244][13] [59][83][95][180][234][123][56][228][186][189]
Topic Distribution	[52][109][143][249][108][51][50][171][112][111][69][114][132][150][191][69]
Location	[187][143][229][75][193][88][201]
Publisher	[75][117][244][228]
Popularity	[172][109][55][82][187][28][75][249][25][28][111][83][88][94][51][75]
CTR	[25]
Recency	[104][109][55][187][28][75][249][171][89][28][111][244][234][201]
Novelty	[50][47]
Dwell Time	[22][55][232][75][249][244][75]
Time Stamp	[75][41][25][43][225][228]
Emotion/Sentiment	[147][148]
Bias	[150]
Knowledge Graph	[84][236]
News/User Graph	[118][141][108][50][192][112][114][152][56][186]
Ontology	[58][16][140][17][67][68][18][30][11][202][142][163][173][48][14][15][44]
Visual Information	[131]

Engagement

Reference paper

News Recommendation Models in literature

Method	Year	Information Used	Model
EBNR [144]	2017	Body	Autoencoder
RA-DSSM [96]	2017	Title+Body	~~Doc2vec+NN~~
Khattar et al. [97]	2017	Title+Body	~~Doc2vec+NN~~
3-D-CNN [98]	2017	Title+Body	Word2vec
WE3CN [90]	2018	Title+Body	2-D CNN
NPA [204]	2019	Title	CNN+Personalized Attention
NRMS [207]	2019	Title	Self-Attention+Attention
NRHUB [206]	2019	Title	CNN+Attention
DAINN [238]	2019	Body	CNN+Dynamic Topic Model
FIM [196]	2020	Title	Dilated CNN
NRNF [209]	2020	Title	Transformer+Attention
FedRec [158]	2020	Title	CNN+Self-Attention+Attention
CPRS [213]	2020	Title+Body	Self-Attention+Attention
UniRec [218]	2021	Title	Self-Attention+Attention
FeedRec [215]	2021	Title	Transformer+Attention
FairRec [221]	2021	Title	Transformer+Attention
EEG [243]	2021	Title+Abstract+Body	CNN+Attention
AMM [240]	2021	Title+Abstract+Body	PLM
RMBERT [81]	2021	Title	PLM
UNBERT [241]	2021	Title	PLM+Attention
PLM-NR [214]	2021	Title	PLM+Attention
SFI [239]	2021	Title	CNN+Attention
TempRec [216]	2021	Title	Transformer
WG4Rec [176]	2021	Title+Word Graph	GNN+Attention
CNE-SUE [136]	2021	Title+Abstract	LSTM+Self-Attention+Co-Attention

Ex. g word2vec

Reference paper

NewsArticle is a Document

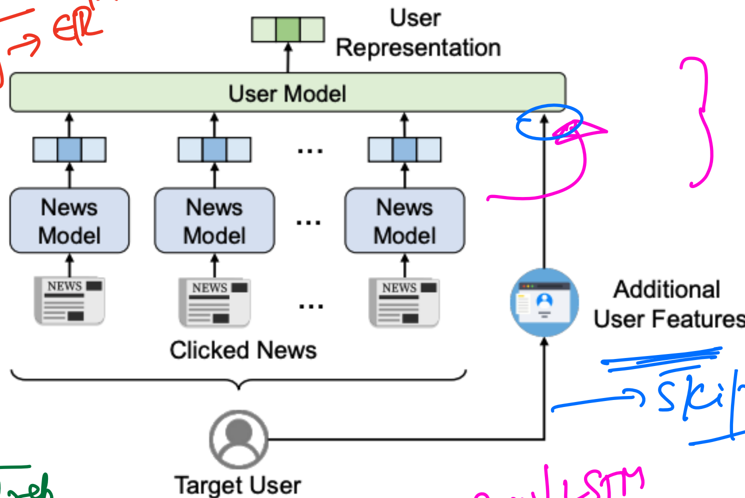
N117 Doc2vec → V117 → $\hat{V}_{117}$

User Modeling

Sentence2vec (Simple:- weighted avg of word2vec of the words)

Sentence:-
I like chocolate icecream
w2v(chocolate) → [ ] → ER
+w2v(icecream)
2
= S2V(Sentence)

Behavior Related



use News Representation to get a user representation

→ Skip-layer

RNN | LSTM

simple user model:
weighted avg of News model rep. that user has clicked!

Reference paper

User 9 → News 7, News 8, News 117, News 200

U_9 → user representation

LSTM

News Rep. from News model for 7! ← V_7

News 7 News 8 News 117 News 200

User Modeling Papers

Method	Year	Information Used	Model
EBNR [144]	2017	News Click	GRU (Fancy LSTM)
RA-DSSM [96]	2017	News Click	Bi-LSTM+Attention
Khattar et al. [97]	2017	News Click	Exponential-decayed Average → weighted avg.
3-D-CNN [98]	2017	News Click	Word2vec
Park et al. [149]	2017	News Click	LSTM
WE3CN [90]	2018	News Click	3-D CNN
DKN [197]	2018	News Click	Candidate-Aware Attention
Gao et al. [49]	2018	News Click	Candidate-Aware+Multi-Head Attention
Saskr [24]	2019	News Click	Self-Attention+Candidate-Aware Attention
NAML [203]	2019	News Click	Attention
TANR [205]	2019	News Click	Attention
NRMS [207]	2019	News Click	Self-Attention+Attention
Liu et al. [119]	2019	News Click	Time-decayed Average
DAN [248]	2019	News Click	LSTM+Self-Attention+Candidate-Aware Attention
KRED [121]	2020	News Click	Attention
TEKGR [103]	2020	News Click	Candidate-Aware Attention
FIM [196]	2020	News Click	3-D CNN
FedRec [158]	2020	News Click	Self-Attention+Attention+GRU
SentiRec [212]	2020	News Click	Transformer+Attention

Reference paper

Breakout #3

How much data to use?

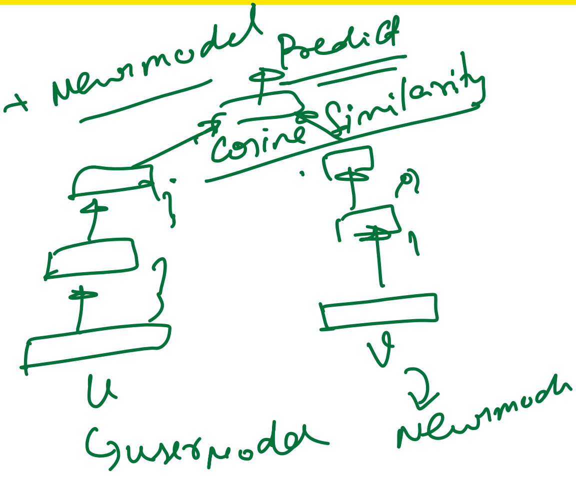
Let's say you have a user who has browsing news articles on a news aggregator webpage for a couple of years. How much of this browsing data would you consider using for your news recommender model (e.g. all of it, some of it, select portions, higher level abstractions, etc) and why? What kinds of features are important to capture from the news browsing history data? What are some unique challenges to user modeling that come up in news recommender systems?

Joint user/news modeling

1. Baseline model?

1) NN + user model + news model

2) SVD (simple) → sparsity / imputation issues



Joint user/news modeling

1. Baseline model?
2. Can cast as a binary classification problem (where click is positive and negative sampling gives negatives)

↳ will have noise

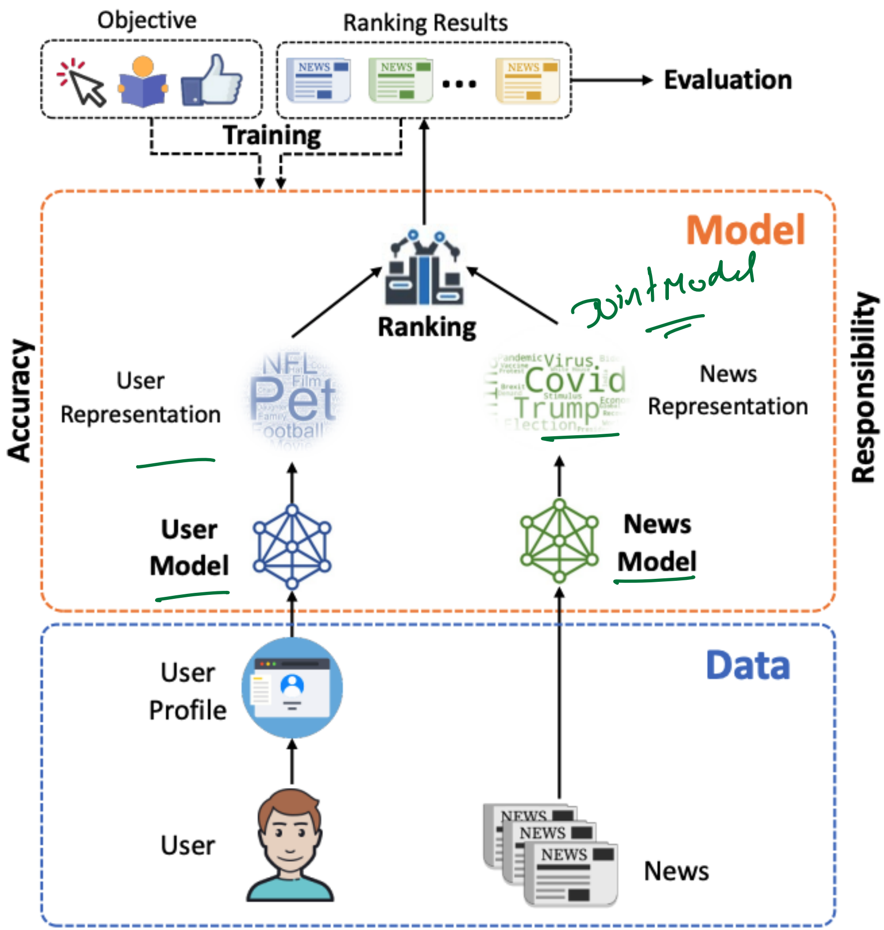
Joint user/news modeling

1. Baseline model?
2. Can cast as a binary classification problem (where click is positive and negative sampling gives negatives)
3. As we want to model user clicking news articles - Both user and news representations (models) get learned jointly / share latent space

Joint user/news modeling

1. Baseline model?
2. Can cast as a binary classification problem (where click is positive and negative sampling gives negatives)
3. As we want to model user clicking news articles - Both user and news representations (models) get learned jointly
4. Has advantage of coordinating representations between the news and user models better in the learning process vs learning them separately

Joint user/news modeling



User Modeling unique dimensions

1. Short shelf life of news
2. Fine-grained interaction of users with topics
3. User can be thought of as a document!
4. More Temporal diversity as compared to movie recommendations

→ Contrast with product recommendations

Metrics for News Recommendations

Metrics for News Recommendations

1. Click based metrics → P/R/F-score/AUC
2. Engagement based metrics → Dwell Time
3. Online vs Offline metrics

4. Ranking metrics → Next Lecture

Assignment 3 - Kaggle Contest!

1. Assignment on Kaggle on news recommendations! **Due Date: August 10**

Assignment 3 - Kaggle Contest!

1. Assignment on Kaggle on news recommendations! **Due Date: August 10** (2 weeks from today)
2. Your submissions will be ranked on leaderboard by Kaggle (supposed to be fun and motivating!)

Assignment 3 - Kaggle Contest!

1. Assignment on Kaggle on news recommendations! **Due Date: August 10**
2. Your submissions will be ranked on leaderboard by Kaggle (supposed to be fun and motivating!)
3. Try out baseline recommenders (SVD based)

Assignment 3 - Kaggle Contest!

1. Assignment on Kaggle on news recommendations! **Due Date: August 10**
2. Your submissions will be ranked on leaderboard by Kaggle (supposed to be fun and motivating!)
3. Try out baseline recommenders (SVD based)
4. Try out any of the NN based user and news models for better model performance!

Assignment 3 - Kaggle Contest!

1. Assignment on Kaggle on news recommendations! **Due Date: August 10**
2. Your submissions will be ranked on leaderboard by Kaggle (supposed to be fun and motivating!)
3. Try out baseline recommenders (SVD based)
4. Try out any of the NN based user and news models for better model performance!
5. Starter notebook provided for helping process the data and understand it better

Next Class

1. Ranking vs Recommendations problem
2. Ranking example applications
3. Ranking Loss functions
4. Learning to Rank (classic ML problem in Search)
5. Ranking metrics (different from recommendation metrics)