Evaluation Approaches

- **Industry outcome**
  - Add-on sales
  - Click-through rates
- **In research**
  - Offline: To anticipate the above beforehand
    - No actual users are involved and an existing dataset is split into a test and a training set
    - Using the ratings in the training set, predict the ratings in the test set
    - Predicted ratings are compared with ratings in the test set using different measures
    - In K-fold cross validation (a common cross validation technique), the data set is partitioned into K equal-sized subsets: one is retained and used as the test set, the other subsets are used as training set. This process is repeated K times, each time with a different test set.
  - Online: User satisfaction

Evaluation Metrics

- Accuracy Metrics
  - measure how well a user's ratings can be reproduced by the recommender system, and also how well a user's ranked list is predicted
  - 3 kinds of accuracy metrics
    - Predictive
    - Classification
    - Rank
- Other metrics:
  - Coverage, Confidence, Diversity, Novelty and Serendipity

Predictive Metrics

- measure to what extent a recommender system can predict ratings of users.
- useful for systems that display the predicted ratings to their users.
- MAE = (|0|+|1|+|3|+|0|+|− 2| + |0| + |2|)/7 = 1.143

$$MAE = \frac{1}{|B_i|} \sum_{b_k \in B_i} |r_i(b_k) - p_i(b_k)|$$

	Ranking		Rating	
Item	**User**	**RS**	**User**	**RS**
A	1	1	5	5
B	2	5	4	3
D	3	4	4	4
G	4	6	4	2
E	5	3	3	5
C	6	2	2	5
F	7	7	2	2

Classification Metrics

- measure to what extent a RS is able to correctly classify items as interesting or not.
- Ignores rating difference

- *Precision: #good items recommended/#recommendations*
  - measures proportion of recommended items that are good

- *Recall: #good items/#all good items*
  - measures proportion of all good items recommended

Rank Metrics
DCG, nDCG for list comparison

- A measure of effectiveness of a web search engine algorithm or related applications
- DCG measures the usefulness, or *gain*, of a document based on its position in the result list
- Two assumptions are made in using DCG:
  - Highly relevant documents are more useful when appearing earlier in a search engine result list (have higher ranks)
  - Highly relevant documents are more useful than marginally relevant documents, which are in turn more useful than irrelevant documents.
- DCG originates from an earlier, more primitive, measure called Cumulative Gain.

Cumulative Gain: CG

It is the sum of the graded relevance values of all results in a search result list.

The CG at a particular rank position p is defined as:
where rel_i is the graded relevance of the result at position i.

$$\mathrm{CG_p} = \sum_{i=1}^{p} rel_i$$

CG Example

$$D_1, D_2, D_3, D_4, D_5, D_6$$

the user provides the following relevance scores:

$$3, 2, 3, 0, 1, 2$$

$$\mathrm{CG_p} = \sum_{i=1}^{p} rel_i = 3 + 2 + 3 + 0 + 1 + 2 = 11$$

does not account for document ordering.

Discounted Cumulative Gain: DCG

DCG is that highly relevant documents appearing lower in a search result list should be penalized as the graded relevance value is reduced logarithmically proportional to the position of the result.
The discounted CG accumulated at a particular rank position is defined as:

$$\mathrm{DCG_p} = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2(i)}$$

No theoretical justification for using a logarithmic reduction factor other than it produces a smooth reduction.
An alternative formulation of DCG places stronger emphasis on retrieving relevant documents:

$$\mathrm{DCG_p} = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{\log_2(i+1)}$$

DCG Example

$D_1, D_2, D_3, D_4, D_5, D_6$

the user provides the following relevance scores:

$3, 2, 3, 0, 1, 2$

i	rel_i	$\log_2 i$	$\frac{rel_i}{\log_2 i}$
1	3	0	N/A
2	2	1	2
3	3	1.585	1.892
4	0	2.0	0
5	1	2.322	0.431
6	2	2.584	0.774

So the DCG_6 of this ranking is:

$$\mathrm{DCG_6} = rel_1 + \sum_{i=2}^{6} \frac{rel_i}{\log_2 i} = 3 + (2 + 1.892 + 0 + 0.431 + 0.774) = 8.10$$

Normalized DCG

$$\mathrm{nDCG_p} = \frac{DCG_p}{IDCG_p}$$

Search result lists vary in length depending on the query.

Comparing a search engine's performance from one query to the next cannot be consistently achieved using DCG alone.

The cumulative gain at each position for a chosen value of should be normalized across queries.

Ideal DCG (IDCG) at position is obtained by sorting documents of a result list by relevance, producing the maximum possible DCG till position p.

nDCG Example

$$D_1, D_2, D_3, D_4, D_5, D_6$$

the user provides the following relevance scores:

$$3, 2, 3, 0, 1, 2$$

$$3, 3, 2, 2, 1, 0$$

The DCG of this ideal ordering, or *IDCG*, is then:

$$\mathrm{IDCG_6} = 8.69$$

And so the nDCG for this query is given as:

$$\mathrm{nDCG_6} = \frac{DCG_6}{IDCG_6} = \frac{8.10}{8.69} = 0.932$$

Online Evaluation: User Studies

- Traditionally small-scale controlled experiments: at best 50 subjects
- Large-scale controlled experiments using crowdsourcing such as Amazon Mechanical Turk

Mechanical Turk Summary

- **Provide a "crowd-sourcing" marketplace where**
  - requesters (i.e., individuals or institutions who have tasks to be completed)
  - workers (i.e., individuals who can perform the tasks in exchange for monetary reward) can come together.
- **A platform where the tasks (i.e. HITs) are**
  - hosted and executed, money is transferred securely
  - the reputation of workers and requesters is tracked
- **The simplest HIT often presented as**
  - a web form, where the worker answers the questions on the form
  - AMT transmits the answers to the requester for further analysis
- **The requester can also specify certain criteria that a worker must satisfy in order to perform the task.**
- **A single user can be limited to perform at most x HITs from each group, ensuring user diversity**