**Multi-agent Social Choice for Dynamic Fairness-aware Recommendation**

An implementation of SCRUF-D

(Social Choice for Recommendation under Fairness - Dynamic)

**Background:**   
Recommender systems are personalized machine learning systems that help users find compelling content in large corpora. They often operate as two-sided platforms:   
Consumers - receive recommendations & possibly act on them  
Providers - create and/or provide items that may be recommended  
  
There are three stakeholders (consumers, providers, system) of a recommender system and fairness concerns may derive from any of them, and the concerns may need to be balanced against each other.   
  
**Types of Fairness:**  
Fairness is a contested concept, which can be grounded in specific contexts and specific stakeholders. Many of these fairness ideas will possibly be in tension with each other and certain tradeoffs will need to be established. Some of the types of fairness concerns are outlined below:  
  
a) Representational Fairness: This fairness attribute is concerned with how outputs of a recommender system operate to represent the world and the individual classes within it.   
  
b) Provider-side fairness: Enhance the distribution of recommendations across providers  
  
c) Group Fairness: Fairness across the outcomes relative to predefined protected groups  
  
d) Individual Fairmness: Each individual user has an appropriate outcome and assumes that users with similar profiles should be treated the same.   
  
e) Additional concern: consulting the institutional mission and its internal and external stakeholders   
  
**Logics of Fairness:**  
Moulin (welfare economist) identified four logic of fairness  
  
a) Exogenous Right: some external constraint on the system. E.g. regulatory  
  
b) Compensation: observed harm or extra costs incurred by one group over another  
  
c) Reward: resources may be allocated as a reward for performance  
d) Fitness: resource should go to those best able to use it (efficiency)  
  
**SCRUF-D Architecture:**

SCRUF-D is a model to formalize multistakeholder fairness in recommender systems as a two stage social choice problem. Therefore, recommendation fairness is a combination of an allocation and aggregation problem, integrating fairness concerns and personalised recommendation provisions to obtain new recommendations.

**Limitations addressed by SCRUF-D:**  
a) Multiple fairness concerns will be active at any one time while being relatively unrestricted in form  
  
b) It is a dynamic framework where decisions are made in the context of historical choices and results.

**Fairness Agents:**  
  
a) Evaluation: where *i* is the agent  
Input: history of the system’s actions  
Output: value in the range of [0,1], where 1 = maximally fair, 0 = totally unfair  
  
b) Compatibility: where *i* is the agent  
Input: user profile *w* and associated information  
Output: value in the range of [0,1], where 1 = most compatible user & context, 0 = least compatible user & context

c) Preference:  
Input: Items  
Output: Preference score in (larger value indicates greater preference)

**Recommendation Process:**  
  
a) Generation of recommendation lists when a user arrivces

b) Allocation Phase: determines which fairness agents will be active in responding to a gioven recommendation opportunity.  
  
Output: set of non-negative weights , summing to 1, over the set of fairness agents  
(indicates the extent of each fairness agent)  
  
  
c) Choice Phase:   
Input: Active (non-zero weighted) fairness agents and their weights   
Output: Final list of recommendations

**Formal Description**

1. Users or context: U = {, , … }  
2. Items: V = {, , … }

3. U ; where j = user/context number

V ; where i = item number

4. K-dimensiona; feature vector

5. Set of categorical features each with a finite domain

6. Sensitive features and Sensitive user profiles are associated with one or more fairness agents and belong to a certain protected class.

7. Predicted Rating

8. Recommendation List generated for user/context , = < {, }, … {, } >

Sorted Recommendation List (produces a ranking over the set of items for that user),

sort ( (w, V))

9. For ease of exposition, score all items: un-rated items gets assigned 0 in the Utility matrix

10. **Fairness Agents:**  
Fairness concerns map directly onto agents   
(i) Agents take into account the current state of the system  
(ii) Agents voice their evaluation of how fair;y the overall system is currently operating  
(iii) Agents deduce their compatibility for the current recommendation opportunity  
(iv) Agents outline their preference for how to make the outcomes more fair

Fairness agent, , = {}

a) Fairness Metric denotes how fair recommendations have been so far: ( , )   
 : choice history  
: allocation history

b) Compatibility Metric denotes how compatible the fairness agents believes they are for the user w: ()

c) Ranking function denotes the preferences of the fairness agents: (w, V) {, }