Строение нейронной сети #2

В моей модели учитываются 3 категории абонентов: позитивные, нейтральные, негативные, «позитивные» - это такие пользователи кто часто и много пользуется мобильными услугами и кто хочет развиваться в потреблении контента, «нейтральные» - это такие пользователи, которых полностью устраивает их текущий оператор и тариф, «негативные» - это те, кто желает покинуть компанию. Поэтому у модели №2 на выходе 3 «softmax» нейрона, которые позволяют определить вероятность отношения «абонента» к одной из вышеперечисленных категорий. Для решения подобных задач, была создана «глубокая нейронная сеть», т.к. состоящая из:

* входной «полносвязный» слой с 45 нейронами и активацией «sigmoid»
* скрытый «полносвязный» слой с 25 нейронами и активацией «sigmoid»
* выходной «полносвязный» слой с 3 нейронами и активацией «softmax»

Preprocessing

(Общий)

Программа построчно (для целостности данных) читает переделанный из *.xlxs* в *.csv* файл, проверяя присутствуют ли *nan, null* или *inf* в данной строке, если присутствуют, то эта строка не добавляется. После чего сохраняет запись в *Record()* (класс записей) и помещает эту запись в список. После чего этот список перебирается и на его основе создаётся словарь, ключами которого служат *subs\_id,* а значениями списки классов *Record().* Такая архитектура позволяет по *subs\_id* обратиться к данным любого пользователя (находящегося в словаре).

(Для модели «глубокая нейросеть»)

После этого программа проходится по всем *subs\_id* и их значениям – спискам, в каждом элементе списка программа берёт поле *tariff\_id* и составляет из них *set()* – набор уникальных тарифов, и если он больше 1, то этот пользователь менял свой тариф. После того, как стало известно, какие абоненты меняли тариф, их *subs\_id* помещаются в отдельный список - *positive\_id*, причём убираются те *subs\_id*, которые помечены как «покинувшие» оператора, остальные же, в зависимости от значения колонки «MNP\_OUT», помещаются в *neutral\_id* и *negative\_id*. Далее из этих списков пользователей формируется общая обучающая выборка по такому признаку – берутся по 3 записи на каждого абонента, от туда берутся поля классов *Record()* , то есть берутся ***одинаковые*** поля к примеру «SMS\_IN\_CNT», для всех трёх записей этого абонента и они нормализуются ***между собой***, для того, чтобы нормализация была адекватной, грубо говоря, что бы не сравнивать время с балансом. Затем эти «тройки» преобразуются в один список, который и подаётся на вход модели.