مراحل انجام شده:

داده دارای ۴ ستون اصلی به نام های apps و games و queries و در نهایت سال تولد بود. ستون هدف هم ستون gender بود. داده NaN در هیچ ستونی به جز ستون birth_year وجود نداشت.

در ابتدا در مدل ها ستون birth_year را حذف نکردم و از میانگین سال تولد برای پر کردن مقادیر NaN استفاده کردم.

ستون gender هم به صورت عددی تبدیل کردم.

اما مشکل اصلی لیست بودن ستون های apps, games و queries بود. با استفاده از دستورات به فرم:

```
df = pd.concat([df.pop('queries').apply(pd.Series), df], axis=1)
```

ستون هایی دارای مقدار لیست بودند را تبدیل به چند ستون جداگانه کردم. با توجه به تکرار بسیار کم خیلی از یکسری آی دی ها داده در برخی از ستون ها با مقدار بسار زیادی Nan وجود داشت. این ستون ها را با threshold هشتاد درصد حذف کردم.

اما در هنگام ترین کردن مدل ها نتایج بسیار عجیب بود به همین دلیل با استفاده از یک شبکه عصبی بررسی را انجام دادم و در نهایت به این نتیجه رسیدم که مشکل از داده تغییر یافته بود. تصمیم گرفتم تبدیل لیست یه ستون را خودم دستی انجام بدهم به این صورت که هر ستون نمایشگر آی دی app یا game و یا query باشد. با این ترتیب مقادیر هر یک از این ستون ها میتواند صفر یا یک باشد(استفاده کردن و نکردن)

البته به دلیل زیاد بودن آی دی ها با threshold ۵ درصد نسبت به سایز دیتاست برخی از آی دی ها را حذف کردم.

در نتیجه به ۲۹۱ فیچر رسیدم و بر اساس آن train را انجام دادم. از مدل های زیر استفاده کردم:

۱- logistic regression

۲- decision tree

۳- random forest

۴- KNN

۵- یک شبکه عصبی Dense

در این بین logistic regression بیشترین accuracy را داشت اما بالاترین balance_accuracy با حدود ۷۴ درصد و دقت 82 برای شبکه عصبی بود.