

使用技術

Python, Django, JavaScript, jQuery, Bootstrap5

マッチングについて

対戦希望者を詰める用のテーブルをデータベースに作っておく。[対戦相手を探す]を押した際にAjaxでpost requestを投げて、そのユーザーをテーブルに追加し、同じテーブルからレートに近い範囲(現状は±10000)から相手をランダムに決定。相手が見つかったら相手と自身を対戦希望者のテーブルから自身を削除する。対戦相手は再帰関数で0.5秒ごとに探し、探している途中で自身が対戦希望者のテーブルから削除されていたら(相手側の処理で)終了する。対戦相手が見つかったらやり取りをする部屋を作成し、そのURL情報をjsonでブラウザに返す。ブラウザはそのURLを受け取り、そこに移動する。

レート計算について

イロレーティングをデュエルリンクスのKCぽくなるようにアレンジしている。

イロレーティングでは、 $\text{Player } i$ と $\text{Player } j$ が対戦した際の、 $\text{Player } j$ の期待勝率を e_{ij} , $\text{Player } i$ のレートを r_i としたときに

$$e_{ij} = \frac{1}{1 + 10^{\frac{r_j - r_i}{C}}}$$

が成り立つ。 C は定数で、2人のレート差 $r_i - r_j$ が C の時に $\frac{e_{ij}}{e_{ji}} = 10$ が成り立つような値を選ぶと良い？

言い換える、期待勝率に10倍の差が出てくるようなレート差を C とする。今回はとりあえず40000にしている。

実際のレート更新は、

$$\Delta r_i = K(g - e_{ij})$$

で行っていて、 K はレート増減幅の基準となる定数で、今回は1000としている。 g は勝敗結果を代入する変数で、 $\text{Player } i$ が勝った場合は1を、負けた場合は0を、引き分けの場合は $\frac{1}{2}$ を代入する。