MIXING OBJECTIVE AND SUBJECTIVE WITHIN A SINGLE JUDGMENT

A way to take advantage of the engine is by training such that your code generates default, sparse judgments based on algorithms you devise. Your default code is considered the objective part of the judgment which should be based upon your app's input data.

Yet, suppose there are some aspects of your app's judgment which come from a 'feel' you have for the judgment at hand in the overall case, at the time you see the judgment, and you can't readily encode these aspects in an algorithm. For this situation, when creating training cases for the engine, add the extra keywords to your training judgment following the default, sparse judgment.

The result is that the app's training judgment has then an objective, default component and a subjective component.

Later, after the engine has been trained, in evaluation your app outputs its sparse judgment given some input. This is submitted to the engine as a System 2 query. The engine evaluates given the input data, and previously trained subjective keywords are appended to the returned judgment.

In this way, both the calling app and the engine contribute to the result – the calling app objective and the engine subjective aspects of the judgment.

Alternative terms can be input by the user in the judgment before a query. In this fashion functionality enabling constant objective characteristics and varying subjective characteristics of an output judgment are possible.