

自動車が人に脅威を与える状況としては、

- (1) 自分が運転席で制御権を持っている状況
- (2) 他者が運転席で制御権を持っている状況
- (3) 他の自動車が通行者に脅威を与える状況

と、いくつか考えられます。どれも、オントロジー **danger** が発火されるわけで、基本的に「交通の脅威」というようなコンセプトにまとめられているでしょう。でも、その他のオントロジー (**have_control,not_have_control**) とかミラーリングで、他者が制御を持っていることを把握できる状況だったりして、「交通の脅威」もさらに下位のコンセプト群に分けられていて、それらが状況によって発火するでしょう。

発火は複数のコンセプトが同時に行います。歩行者のコンセプトが発火するでしょう。道路の状況に関するコンセプトが発火するでしょう。脅威のコンセプトが発火するでしょう。自動車の運転を司るコンセプトも発火しているでしょう。自分が運転席で制御しているという状況ですと、現行の行動の管理のところに、自動車運転のコンセプトが登録され、発火しているでしょう。歩行者が脅威になっていると状況は、自動車が走行しているときに、常にオブジェクトチェック機能で、監視していて、いざという時に、脅威コンセプトに対象と危険の種類とが表示されるでしょう。それをもって、危険回避コンセプトを連想によって、発火します。そこから、自動車運転のコンセプトにコマンドが発せられます。自動運転コンセプトは危険回避の運転に切り替えて、自動車の動きを制御していきます。その中身も、プロダクションシステムのもとの発想が行われます。一連の作業の脅威回避のコンセプトが行動管理リストに登録されていて、その行動の実現が実行されているか監視もしていつ、行動が途中で消えることが無いようにします。目標が達成されたら、脅威回避のコンセプトを非活性にします。

LifeLog 管理に発火として登録されたコンセプトはオントロジーからサブシステムに登録されるべきでしょう。そのサブシステムの持っている連想関係から次々に関連事項を発火していきます。サブシステムの他に、現行の行動と予定の行動をメインシステムに登録していますから、それに関連するコンセプトから次第に連想して発火していくことにもなります。ものすごい、発火の連鎖が起きることになります。コンセプトは短い行動要素の連鎖としてもあるでしょうし、大きな制御の塊のこともあるでしょう。プラントのモデル予測制御ですと、それなりに大きなシステムになっています。それでも、突発事項があったときにはこうするというようなプロセスを本体とは別のコンセプトでもつこともあるでしょう。いずれも、重み付投票で、プロセスのスイッチは切りかえられるわけです。いずれにしても、現在の状況というものを共有したデータとして利用できれば、制御の移行は

スムーズに行くでしょう。

ちょっとした作業でも、ずいぶん複雑な処理を組み合わせねばならないことが分かります。連想とか発想とか行動を起こすまでの流れは、複雑なネットワークになります。ノードはコンセプトです。複雑だから、コンセプトとコンセプト間の関係（2項関係）を管理していくことで、全体のカオス的狀況を管理していく必要があります。このコンセプトとコンセプト間の関係を構築していくのが機械学習です。それは人手による構築は困難なものを解決するためのものです。

おわり