

無理問答生成器

西村 郷

無理問答とは

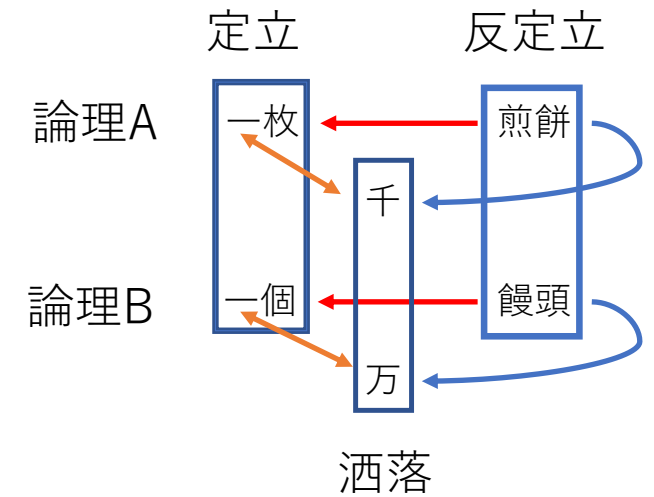
「一枚でも煎（千）餅とは、これ如何に」

「一個でも饅（万）頭と言うが如し」

ある言葉（一枚）を持ってきて、それが現実には指し示すような言葉（煎餅）を持って来るのだが、実はその言葉の中には前者と矛盾するような言葉（一枚なのに千）が含まれている。

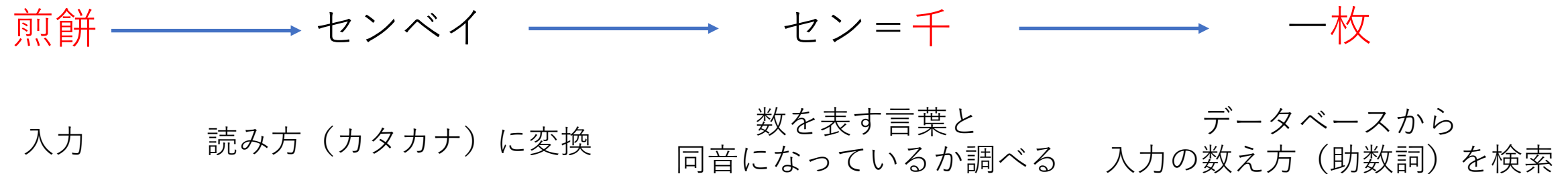
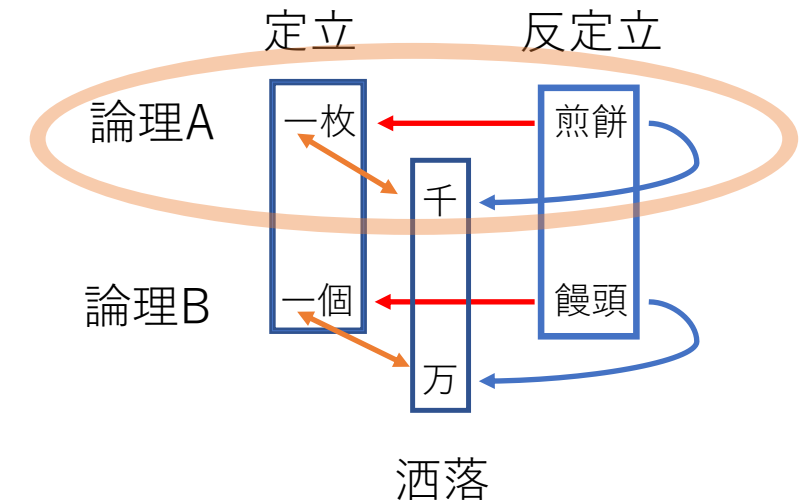
確かにそれらは矛盾を起こしているのだが、同じような事物（一個、饅頭、一個なのに万）についても同じような矛盾が起こっているのだから、それは問題ないのだ、と言いつける言葉遊びの一種である。

生成器作成への戦略



- この無理問答の意味的な構造を整理すると、右のような図になる。
- ポイントは、論理Aと論理Bで対応する部分、つまり定立、反定立、洒落のグループの中の言葉同士は意味が似通っていることである。
- すなわち、論理Aさえ出来上がれば、適当な言葉を一つ論理Bに与えて論理Aの言葉間の構造を移し替えることで論理Bも作ることができる
 - 例えば、右上の図の論理Aが完成したとする。論理Bの反定立は「饅頭」が良かった時、論理Aの構造（**赤矢印**）を用いて論理Bの定立は「一個」で良いことが分かる。あとは、「饅頭」という言葉の駄洒落（**青矢印**）が「一個」と矛盾（**橙矢印**）してさえすれば、問答が完成したことになる。
- よって生成器は、論理Aを作るものと、それからの出力を入力とする論理Bを作るものとに分けられる。

論理A生成器



- 定立と反定立の関係性は必ずしも物とその助数詞である必要はないのだが、とりあえず最も無理問答を生成しやすいこのパターンに絞った。

論理B生成器

論理A生成器からの入力

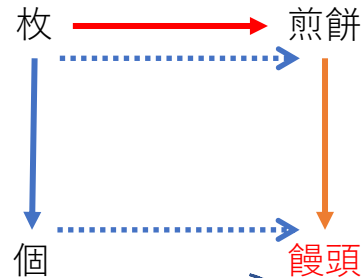
一枚でも煎餅（千）とは、これ如何に

本、**個**、匹、人、...

1 : 意味の類似する
言葉を見つける

個 - 枚 + 煎餅

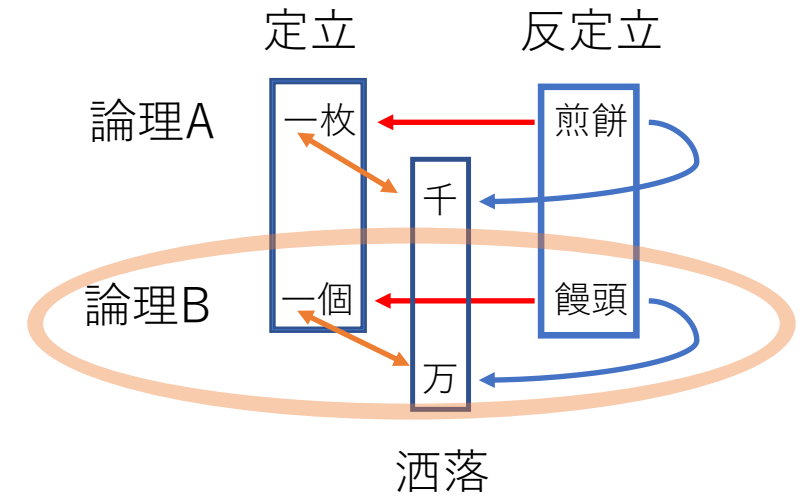
→ **饅頭**



饅頭 = マンジュウ、マンジュ、ンジュウ、...、マン、ンジ、...
= **万**

千

意味が似ている = 終了



2 : Word2Vecを用い、意味のベクトル演算を行う

3 : 見つけた言葉から、論理Aの駄洒落と
似た意味になる駄洒落を見つける

- この図では定立から論理Bを組み立てているが、実際のプログラムでは反定立からも組み立てを行っている。

出力例

一輛なのに戦車(千)とはこれ如何に

一輛なのにM4中戦車(千)と言うが如し score: 0.7562747001647949
一輛なのにV号戦車パンター(千)と言うが如し score: 0.7464680671691895
一輛なのにIV号戦車(千)と言うが如し score: 0.7154760956764221
一隻なのに潜水艦(千)と言うが如し score: 0.7053159475326538
一輛なのに九五式軽戦車(千)と言うが如し score: 0.7033185958862305
などなど

一枚なのに煎餅(千)とはこれ如何に

一枚なのに饅頭(万)と言うが如し score: 0.8412349224090576
一円なのに海鮮(千)と言うが如し score: 0.5618506073951721
一mmなのに温泉卵(千)と言うが如し score: 0.529010534286499
一桁なのに線香(千)と言うが如し score: 0.4936918318271637
一個なのに中華まん(万)と言うが如し score: 0.4016512632369995

一匹なのに秋刀魚(三)とはこれ如何に

一段_(単位)なのに実録忠臣蔵(六)と言うが如し score: 0.6309561729431152
一畝_(単位)なのに保戸野八丁(八)と言うが如し score: 0.6196873188018799
一隻なのにブルターニュ_(戦艦)(二)と言うが如し score: 0.6193463802337646
一箇なのに一乗寺(一)と言うが如し score: 0.6163501143455505
一箇なのに一乗寺(一條)と言うが如し score: 0.6163501143455505
などなど

課題と考えられる原因

- 訳の分からない論理を頻繁に作ってくる
 - 使用しているモデルによる表現が悪いのかもしれない（といってもこれが手に入れられるもので最高のもののはずだが）
 - 答えの選別方法にもっと工夫がいるのかもしれない
 - 論理Aで取ってくる言葉がBに影響する
- 実行時間がとてもつもなく長い（1時間半かかったケースも）
 - A生成器で物の数え方を、B生成器で駄洒落の導出（一番のボトルネック）と答えの選別でネットに接続しているところで主に時間を食っているので、これを別の方法で表現する必要がある
- 作る論理（A）がワンパターン
 - 他の論理の形とその作り方も対応できるようにしたい

関数一覧

- mecab_parser
 - 文字列を形態素に分割する
- utf8_encode
 - 入力された日本語をUTF8にエンコードする
- katakanize
 - 入力された言葉をカタカナに変換する
- hiraganize
 - 入力された言葉をひらがなに変換する
- partDivider
 - 入力を自身の言葉の部分集合に変換する
- kanjize_slow
 - リスト中の文字を一語ずつ漢字に変換する
- kanjize_fast
 - リスト中の文字を一度に漢字に変換する
- chooseRealWords
 - リストから日本語として意味の通る言葉を選別する
- get_syare
 - 入力の駄洒落を見つけてくる
- get_similar_word
 - 入力と似た意味の言葉を見つけてくる
- cos_sim
 - 二つの言葉ないしベクトル間のコサイン類似度を計算する
- search_hits
 - 入力をグーグルで検索した時の検索件数を返す
- search_page
 - 指定のページを閲覧し、入力の助数詞を返す
- get_suusi
 - 入力の助数詞を返す
- upper_half
 - 論理Aを生成する
- lower_half
 - 論理Bを生成する

呼び出し関係

