допустим, что у тебя есть варианты разбиения на морфемы, и ты получила список, где собственно морфемы заменены на возможные их типы

[[мам, а, ми], [мам, ам, и], [мам, ами]]

[['R'], ['s', 'f'], ['s']]

[['R'], ['s', 'f'], ['s', 'i', 'f']]

[['R'], ['f']]

[[('R', 's', 's'), ('R', 'f', 's')], [('R', 's', 's'), ('R', 's', 'i'), ('R', 's', 'f'), ('R', 'f', 's'), ('R', 'f', 'i'), ('R', 'f', 'f')], [('R', 'f')]]

вот три варианта разбиения

я беру каждый из них, собираю возможные типы морфем в свой массив.

собрав все типы (т.е. три массива с типами, соответствующие трем морфемам в первом разбиении), я просто запускаю product и получаю список векторов, как в последней строке

по-моему, это самый простой вариант, но лучше сама скажи, что здесь смущает

[[('R', 's', 's')], [('R', 's', 's'), ('R', 's', 'f')], [('R', 'f')]]

ну по мере того, как продакт их выдают, я собираю их в массив

затем я прохожусь по элементам массива, создаю объект класса, который отвечает за конечный автомат. он, в свою очередь, проходится по внутренним элементам, а потом говорит, возможна ли такая цепочка или нет. если возможна, то она сохраняется.

но сейчас я понял, что проще запускать конечный автомат по мере работы продакта, а не сохранять все лишний раз

здесь у меня используются только два класса

в классе морфем заданы атрибуты: собственно строка с морфемой, конечная позиция, типы

кажется, можно обойтись и без него, но тогда еще на самом первом этапе нужно как-то отдельно хранить позиции морфем, что мне кажется сложнее

в классе конечного автомата есть атрибуты: таблица переходов (просто отдаю объекту заранее написанный словарь) и нынешнее состояние, которое по умолчанию равно 'start'.

еще там есть метод, который берет последовательность элементов и выдает, возможна ли она. Добров сказал там, правда, немного все упростить. и вообще ему надо как-то дуги в конечном автомате учитывать, так что я не знаю, что еще там сделаю

ну то есть в слове "мамами" морфема "мам" обладает строкой "мам", конечной позицией 4 (где начинается следующая морфема) и типом R