

1. *Какие виды слоев используются в АРТ-сетях? Какой функционал они выполняют?*

Слой сравнения - число нейронов на нем фиксировано и равно размерности входного вектора. На данном слое осуществляется сравнение значения слоя распознавания и текущего входного образца. Для этого входной вектор преобразуется в вектор S , который передается в весовую матрицу действительных чисел. На выходе каждого нейрона слоя распознавания формируется ожидаемый вектор V (типичный представитель данного класса) и из них выбирается образец с наибольшим соответствием. Затем S и V сравниваются с так называемым параметром соответствия. Если схожести не обнаружено, то на слое распознавания создается новый нейрон.

Слой распознавания (выходной слой/соревновательный слой) - число нейронов переменное, каждому нейрону соответствует некоторый класс. Первоначально на слое есть только один нейрон. Данный слой сопоставляет каждому входному вектору класс. Если подходящего класса нет, образуется новый класс (к слою добавляется новый нейрон).

2. *Какую метрику можно использовать для сравнения двух текстов?*

Метрик много. Одним из вариантов можно использовать Jaccard Similarity. Метод рассматривает предложения как набор слов. Статистика: Размер пересечения поделить на размер объединения двух множеств.

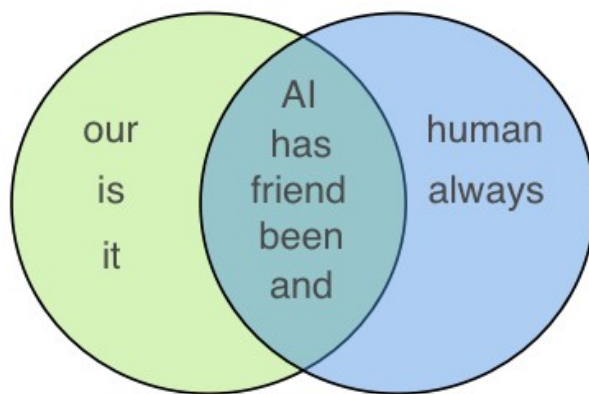
Пример:

Предложение 1: AI is our friend and it has been friendly

Предложение 2: AI and humans have always been friendly

Чтобы вычислить близость предложений, необходимо сначала избавиться от словоформ: friend & friendly будем считать за одно слово, как и has & have.

Диаграмма Венна для данных множеств приведена ниже.



$5/(5+3+2) = 0.5$ значение метрики в данном случае.

Данная метрика обладает определенными недостатками: не учитывается контекст, и с увеличением размера текста число общих слов так же будет расти. И тем не менее, такую метрику *можно* использовать для измерения расстояния между текстами.

3. Для чего нужна функция *vectorize*?

Идея векторизации присутствует и в других языках программирования: и в R, и в Octave, например. Суть везде примерно одинакова: пусть имеется функция, которая умеет вычислять некоторое значение, при конкретном значении параметров $f(a,b)$ (параметров, понятно, может быть и больше). И если у нас есть массив значений для a и для b , то по простому, чтобы вычислить функцию для всех пар, необходимо каждый раз писать цикл `for`, что чаще всего бывает неэффективно. Функция `vectorize` позволяет этого избежать. Если мы хотим вычислить \sin от $x=[1,2,3,4]$, то достаточно лаконичного вызова $\sin(x)$ (что было бы невозможно без векторизации в `numpy`). Векторизация в R позволяет использовать параллельность и даже строить регрессионные модели с помощью уже векторизованных функций.