Алгоритм

Обозначения

Пусть:
Функция intersect(A, B) возвращает элементы, которые содерержатся одновременно и в массиве A, и в массиве B.
Функция size(A) возвращает количество элементов, которые содержатся в массиве A.
Функция lcs(A, B) возвращает наибольшую общую подпоследовательность слов в A и B
sent - массив предложений.
sent[i].nGrams - массив n-грамм в i-ом предложении.
classes - массив классов
classes[i].nGrams - массив n-грамм в i-ом классе.
classes[i].sent - массив предложений в i-ом классе.

Algorithm 1 Поиск неточных повторов

```
1:  for i = 1 to size(sent) do
2:      curSent = sent[i]
3:      bestOverlap = 0
4:      bestClass = NULL
5:      for j = 1 to size(classes) do
6:          curClass = classes[j]
7:          curIntersect = intersect(curSent.nGrams, curClass.nGrams)
8:          curOverlap = size(curIntersect) / size(curSent.nGrams)
9:          if curOverlap > bestOverlap then
10:             bestOverlap = curOverlap
11:             bestClass = curClass
12:         end if
13:     end for
14:     if bestOverlap < 0.5 then
15:         Создать новый класс newClass
16:         newClass.nGrams += curSent.nGrams
17:         newClass.sent += curSent
18:     else
19:         bestClass.nGrams += curSent.nGrams
20:         bestClass.sent += curSent
21:     end if
22: end for
```