Algorithm 1 Rule-Based System (RBS)

```
Require: $corpus \leftarrow list(words) \geq 0$
  for $sent \leftarrow example\_system\_transcr$ do
    $sent \leftarrow drop\_duplicate\_char(sent)$
    for $token \leftarrow sent$ do
      for $gold \leftarrow corpus\_1$ do
        if $token$ in $gold$ then
          $gold, subtoken \leftarrow split\_token(token)$
          $sent \leftarrow replace\_token\_in\_sentence(token, [gold, subtoken])$
        end if
      end for
      $list[(gold_1, gold_2)] \leftarrow create\_pairs(corpus)$
      for $pair \leftarrow list[(gold_1, gold_2)]$ do
        $combination \leftarrow pair[0] + pair[1]$
        if $token$ in $combination$ then
          $gold_1, gold_2 \leftarrow split\_combination(token)$
          $sent \leftarrow replace\_token\_in\_sentence(token, [gold_1, gold_2])$
        end if
      end for
      $token \leftarrow replace\_freq\_tokens(token)$
      $list\_and \leftarrow$ ['καὶ', 'καὶ', 'καί']
      for $gold \leftarrow corpus + list\_and$ do
        if $edit\_distance(gold, token)$ == 1 and (token not in $list\_and$) then
          if gold in $list\_and$) then
            if gold not in ($begin/end\_of\_the\_sentence$) then
              $token \leftarrow gold$
            end if
          else if $N$ is odd then
            $token \leftarrow gold$
          end if
        end if
        if $edit\_distance(gold, token) == 2$ and $length(token) \geq 8$ then
          $token \leftarrow gold$
        end if
      end for
      $list\_articles \leftarrow$ ['τὴν', 'κατα', 'τὰ', 'τῶν']
      if token in $list\_articles$ then
        if position(token,gold) in $begin\_or\_end\_of\_token$ then
          $gold, subtoken \leftarrow split\_article(token)$
          $sent \leftarrow replace\_token\_in\_sentence(token, [gold, subtoken])$
        end if
      end if
      if length(token)==1 then
        $sent \leftarrow drop\_token(token)$
      end if
      for $i \leftarrow range(0, len(sent\_tokens) - 1)$ do # R3
        $w1, w2 \leftarrow sent\_tokens[i], sent\_tokens[i + 1]$
        $bigram = w1{+}w2$ # no white space between the consecutive words
        for $g \leftarrow corpus$ do # for each gold word in the corpus
          if $edit\_distance(g, bigram) == 1$ & $w1$ not in {'ο','η','το','τα'} then
            token ←g+‘ ’+w2
          end if
        end for
      end for
    end for
  end for
```