The pseudo code in (PF)Specification-stage

The pseudo code of feature Sim(Q,KBQ)

Input: Query and Question
Output: the result of Sim(Q,KBQ)

```
1.    list1 = semantic(Query),list2 = semantic(Question)    //Extract the semantic block of Query to form list1, extract the semantic block of Question to form list2.
2.    for i, item1 in enumerate(list1) do    //Iterate through list1 to get each semantic block item item1 of the Query
3.        for i, item2 in enumerate(list2) do    //Iterate through list2 to get each semantic block item item2 of the Question
4.            score = synonyms.compare(item1, item2, seg=False)    //Use the Synonyms toolkit to calculate the similarity between item1 and item2 semantic blocks to get the score, seg=False means no more splitting of semantic blocks
              if score > score_max then    //score_max is used to semantically preserve the maximum value of the similarity score
5.                score_max = score
6.            end if
            list.append(score_max)    //To get the score_max of item1 on item2 and save it in the list
7.        endfor
8.    endfor
9.    Sim(Q,KBQ) = sum(list) / len(list)
10.   return Sim(Q,KBQ)
```

The pseudo code of feature Sim(Q,KBA)

Input: Query and Answer
Output: the result of Sim(Q,KBA)

```
list1 = semantic(Query),list2 = semantic(Answer)   // Extract the semantic block of Query to form list1, extract the semantic block of Answer to form list2
for i, item1 in enumerate(list1) do   //Iterate through list1 to get each semantic block item item1 of the Query
    for i, item2 in enumerate(list2) do   //Iterate through list2 to get each semantic block item item2 of the Answer
        score = synonyms.compare(item1, item2, seg=False)   //Use the Synonyms toolkit to calculate the similarity between item1 and item2 semantic blocks to get the score, seg=False means no more splitting of semantic blocks
        if score > score_max then   //score_max is used to semantically preserve the maximum value of the similarity score
            score_max = score
        end if
        list.append(score_max)   //To get the score_max of item1 on item2 and save it in the list
    endfor
endfor
Sim(Q,KBA) = sum(list) / len(list)
return Sim(Q,KBQ)
```

The pseudo code of feature Sim(QE,KBQE)

Input: Query and Question
Output: the result of Sim(QE,KBQE)

```
list1 = Entity(Query),list2 = Entity(Question)      //Extract the entity of Query to form list1, extract the entity of Question to form list2
for word in list1+list2 do      //For each word in list1 and list2, look up the word in cilin_vocab and save the words not included in cilin_vocab to vocab_no_words
    if word not in cilin_vocab then
        vocab_no_words.append(word)
    endif
endfor
for i, item1 in enumerate(list1) do      //Iterate through list1 to get each entity item item1 of the Query
    for i, item2 in enumerate(list2) do      //Iterate through list2 to get each entity item item2 of the Question
        inter = (set([item1, item2]).intersection(set(vocab_no_words)))      //Determine if item1 and item2 exist in vocab_no_words
        if inter then
            score = synonyms.compare(item1, item2, seg=False)      //If it exists,use the Synonyms toolkit to calculate the similarity between item1 and item2 semantic blocks to get the score, seg=False means no more splitting of semantic blocks
        else
            score = ci_lin.sim2018(item1, item2)      //If not, calculate the similarity score of item1 and item2 using the HIT synonym word forest method
        endif
        if score > score_max then
            score_max = score      //score_max is used to store the maximum value of entity similarity.
        endif
        list.append(score_max)      //To get the score_max of item1 on item2 and save it in the list
    endfor
endfor
Sim(QE,KBQE) = sum(list)/len(list)
return Sim(QE,KBQE)
```

The pseudo code of feature Sim(QR,KBQR)

Input: Query and Question
Output: the result of Sim(QR,KBQR)

```
list1 = Relation(Query),list2 = Relation(Question)      //Extract the Relation of Query to form list1, extract the Relation of Question to form list2
for i, item1 in enumerate(list1) do      //Iterate through list1 to get each Relation item item1 of the Query
    for i, item2 in enumerate(list2) do      //Iterate through list2 to get each Relation item item2 of the Question
        score = synonyms.compare(item1, item2, seg=False)      //use the Synonyms toolkit to calculate the similarity between item1 and item2 semantic blocks to get the score, seg=False means no more splitting of semantic blocks
        if score > score_max then
            score_max = score      //score_max is used to store the maximum value of entity similarity.
        endif
    endfor
endfor
Sim(QR,KBQR) = score_max
return Sim(QR,KBQR)
```