Reason assessment algorithm explained

No Author Given

No Institute Given

1 Prevalence Function

The Prevalence Score Function, as outlined in Algorithm 1, is designed to assess the relevance or pertinence of a given key reason in the context of a clinical case (CC) and a knowledge base (KB). The function operates by assigning a score to the key reason based on its presence in the correct and incorrect diagnoses, its occurrence rate, and its presence in the clinical case.

The function begins by initializing the score to zero and setting several boolean variables to false (lines 2-6). It then retrieves the symptoms associated with the correct and incorrect diagnoses from the knowledge base (lines 7-8) and identifies the symptoms present in the clinical case using Named Entity Recognition (NER) (line 9).

The function then checks if the key reason is present in the symptoms of the correct diagnosis (lines 10-15). If it is, the function increments the score and sets the variable *linkedToCorrectDiagnosis* to true. If not, *linkedToCorrectDiagnosis* is set to false.

Next, the function checks if the key reason is present in the symptoms of the incorrect diagnoses (lines 16-20). If it is, the variable *linkedToIncorrectDiagnosis* is set to true. If not, it is set to false.

The function then checks the occurrence rate of the key reason (lines 21-28). If the key reason has a high occurrence rate (more than 70%), the variable *hasHighOccurrenceRate* is set to true, and if it is linked to the correct diagnosis, the score is incremented. If the key reason has a low occurrence rate (less than 30%), the variable *hasLowOccurrenceRate* is set to true.

The function then checks if the key reason is unique to the correct diagnosis or shared with other diagnoses (lines 29-40). If the key reason is unique to the correct diagnosis and has a low occurrence rate, the score is incremented twice. If the key reason is shared with other diagnoses, the score is decremented.

Finally, the function checks if the key reason is present in the symptoms of the incorrect diagnoses but not in the present symptoms (lines 41-48). If the key reason has a high occurrence rate, the score is incremented. Otherwise, the score is decremented.

The final score represents the prevalence of the key reason in the context of the specific clinical case, providing a measure of its relevance or pertinence.

Algorithm 2 Prevalence Function

procedure PrevalenceFunction($KeyReason$, CC, KB)
  $score = 0$
  $uniqueToCorrectDiagnosis = False$
  $sharedToOtherDiagnosis = False$
  $hasLowOccurrenceRate = False$
  $hasHighOccurrenceRate = False$
  $SymptomsOfCorrectDiagnosis = KB(CorrectDisease)$
  $SymptomsOfIncorrectDiagnosis = KB(IncorrectDiseases)$
  $PresentSymptoms = NER(CC)$
  if $KeyReason$ is in $SymptomsOfCorrectDiagnosis$ **then**
    $linkedToCorrectDiagnosis = True$
    score = score + 1
  else
    $linkedToCorrectDiagnosis = False$
  end if
  if $KeyReason$ is in $SymptomsOfIncorrectDiagnosis$ **then**
    $linkedToIncorrectDiagnosis = True$
  else
    $linkedToIncorrectDiagnosis = False$
  end if
  if $KeyReason$ has a high occurrence rate (more than 70%) **then**
    $hasHighOccurrenceRate = True$
    if $linkedToCorrectDiagnosis == True$ **then**
      score = score + 1
    end if
  else if $KeyReason$ has a low occurrence rate (less than 30%) **then**
    $hasLowOccurrenceRate = True$
  end if
  if $KeyReason$ is in $SymptomsOfCorrectDiagnosis$ **then**
    if $KeyReason$ is not in $SymptomsOfIncorrectDiagnosis$ **then**
      $uniqueToCorrectDiagnosis = True$
      $score = score + 1$
      if $hasLowOccurrenceRate == True$ **then**
        $score = score + 1$
      end if
    else if $KeyReason$ is in $SymptomsOfIncorrectDiagnosis$ **then**
      $sharedToOtherDiagnosis = True$
      $score = score - 1$
    end if
  end if
  if $KeyReason$ is in $SymptomsOfIncorrectDiagnosis$ **then**
    if $KeyReason$ is not in $PresentSymptoms$ **then**
      if $KeyReason$ has a high occurrence rate **then**
        $score = score + 1$
      else
        $score = score - 1$
      end if
    end if
  end if
end procedure