Available online at www.sciencedirect.com

ScienceDirect

Procedia Technology 25 (2016) 272 – 279

Procedia Technology

Global Colloquium in Recent Advancement and Effectual Researches in Engineering, Science and Technology (RAEREST 2016)

Evaluation of Scalable Database Driven Reverse Dictionary

Soumya Rajan[a*], Kumary R Soumya[a]

[a]*Department of Computer Science & Engineering, Jyothi Engineering College, Cheruthuruthy, Thrissur, India*

Abstract

While a traditional forward dictionary maps words to their definitions, a reverse dictionary takes a user input phrase with a desired concept, and returns a set of candidate words closely related to the input phrase. This application is significant not only for the general public, but mainly to those who work personally with words. It is also important in the general field of conceptual search. Upon receiving a search concept, the Reverse Dictionary consults the forward dictionary and selects those words whose definitions are similar to the given concept. And thus it is reduced to a concept similarity problem. In this paper, different concept similarity measures are compared and the best among them is proposed. The experimental results shows that the approach used here provides significant improvements in performance level without losing the quality of the result.

Keywords: Dictionaries; Stemming; Stop Words; Pos Tagging; Concept Mining,

1. Introduction and Related Works

When a regular (forward) dictionary maps words to their definitions, a Reverse Dictionary(RD) performs the reverse mapping. That is, given a phrase describing a desired concept, a Reverse Dictionary provides words whose definitions match the entered definition phrase. Thus for an example, when a forward dictionary returns the meaning of the word “sorrow” as “sadness”, a reverse dictionary recommends the user to enter the phrase “a feeling of deep grief” or “a sense of deep pain” as input, and expects to receive the word “sorrow” and probably other conceptually similar words as output.

We usually have words on the tip of our tongue, but we can’t quite remember it. And that is where the problem lies. The category of people mainly affected by this problem is writers, including professional writers, students, teachers, researchers etc. For most people with a certain level of education, the problem is not the lack of knowledge about the meaning of a word, but, being unable to recall the appropriate word at the time of requirement.

doi:10.1016/j.protcy.2016.08.107

The RD solves this widespread problem[1].

1.1. RD Problem Approach

In RD, upon receiving a search concept, it consults the forward dictionary and selects those words whose definitions are similar to the searched concept. These words then form the output of the RD lookup. The problem then reduces to a concept similarity problem (CSP). Particularly in computer science, concept similarity has been dealt by both Information Retrieval (IR) researchers [2], [3] as well as Database researchers [4]. The main hitch here is that the input of the user is not likely to exactly match the definition of a word in the forward dictionary and the response efficiency needs to be similar to that of forward dictionary online lookups. According to a recent Forrester study, end users become impatient if a website takes longer than 4-5 seconds to respond to a request [5].

Most of the studies regarding the similarity of concepts, model concepts as single words. For example works in text classification, examined intensively in [6], cluster similar documents if they contain co-occurring words . It doesn't consider sentences or phrases as such. In existing word sense disambiguation approaches, researchers search for the contextual meaning of a polysemous word (i.e., a word with many meanings) based on nearby words in the sentence where the target word appears. In such case too it considers a single word at a time. This single word emphasis is well identified in the literature [3], [7]. Where as in RD, semantic similarities must be computed between multiword phrases. And this is well addressed in paper[8].

1.2. Concept Similarity Problem

The semantic similarity between concepts is a measure of semantic distance between two concepts which can be estimated by using ontologies. Semantic similarity can be used to identify concepts with common "characteristics". Even though it is difficult to give a formal definition for relatedness between concepts, we can find the relatedness between them. For example, a small child can say that “apple” and “grapes” are more related to each other than “apple” and “tomatoes”. And these pairs of concepts are related to each other and its structure definition is formally called “is-a” hierarchy[9]. Here we examine three semantic similarity methods and choose one among them.

1.2.1. Hirst and St-Onge Measure (HSO)

HSO[10] measure calculates similarity between concepts using the path distance between the concept nodes in the taxonomy, number of changes in direction of the path connecting two concepts and the allowableness of the path. If there is a close relation between meanings of two concepts or words, then the concepts are said to be semantically related to each other. An Allowable Path is a path that does not deviate away from the meaning of the source concept and thus is considered in the calculation of relatedness. Let ‘d’ be the number of changes of direction in the path that relates two concepts C1 and C2, and C, k are constants whose values are derived through experiments. And ‘len’ is the short path relating (i.e minimum number of links) concepts C1 to concept C2.Then similarity function of HSO is formulated as follows:

$$Sim_{HSO}(C1,C2) = C - len(C1,C2) - k*d \tag{1}$$

In brief according HSO, two lexicalized concepts are semantically close if their WordNet synsets are connected by a path which is not too long and which does not change direction too often.

1.2.2. Wu and Palmer(wup)

In Wu and Palmer[11], let C1 and C2 be two concepts in the taxonomy, this similarity measure considers the position of C1 and C2 to the position of the least common sub-sumer(LCS) C, that is the most specific common concept. Several parents can be shared by C1 and C2 by multiple paths. The most specific common concept is the closest common ancestor C (the common parent related with the minimum number of IS-A links with concepts C1 and C2).