Challenges in Computing Semantic Relatedness for Large Semantic Graphs

Teresa Costa
CRACS & INESC-Porto LA, Faculty of Sciences, University of Porto
Porto, Portugal
teresa.costa@dcc.fc.up.pt

José Paulo Leal
CRACS & INESC-Porto LA, Faculty of Sciences, University of Porto
Porto, Portugal
zp@dcc.fc.up.pt

ABSTRACT

The research presented in this paper is part of an ongoing work to define semantic relatedness measures to any given semantic graph. These measures are based on a prior definition of a family of proximity algorithms that computes the semantic relatedness between pairs of concepts, and are parametrized by a semantic graph and a set of weighted properties. The distinctive feature of the proximity algorithms is that they consider *all* paths connecting two concepts in the semantic graph. These parameters must be tuned in order to maximize the quality of the semantic measure against a benchmark data set. From a previous work, the process of tuning the weight assignment is already developed and relies on a genetic algorithm. The weight tuning process, using all the properties in the semantic graph, was validated using WordNet 2.0 and the data set WordSim-353. The quality of the obtained semantic measure is better than those in the literature. However, this approach did not produce equally good results in larger semantic graphs such as WordNet 3.0, DBPedia and Freebase. This was in part due to the size of these graphs. The current approach is to select a sub-graph of the original semantic graph, small enough to enable processing and large enough to include all the relevant paths. This paper provides an overview of the ongoing work and presents a strategy to overcome the challenges raise by large semantic graphs.

Categories and Subject Descriptors

E.1 [**Data**]: Graphs and networks; G.2.2 [**Mathematics of Computing**]: DISCRETE MATHEMATICS Graphs and networks[Path and circuit problems] ; I.2.4 [**Computing Methodologies**]: ARTIFICIAL INTELLIGENCE Knowledge Representation Formalisms and Methods[Semantic networks]; I.2.8 [**Computing Methodologies**]: ARTIFICIAL INTELLIGENCE Problem Solving, Control Methods, and Search[Graph and tree search strategies]

IDEAS'14, July 07 - 09 2014, Porto, Portugal

http://dx.doi.org/10.1145/2628194.2628200 .

Keywords

Semantic similarity, Linked data, Freebase, DBPedia, WordNet

1. INTRODUCTION

The semantic relatedness measures described in the literature use different approaches, but most of them fall in two main types: knowledge base methods (based on their semantic graphs) and distributional methods (based on frequency of word occurrence in corpora) [3]. There are also hybrid approaches that combine the two paradigms. The methodology presented on this paper follows the knowledge base approach.

This paper presents ongoing work aiming at the development of a methodology to define a semantic relatedness measure between two concepts for a given semantic graph, without requiring knowledge of its domain. It uses a family of semantic relatedness algorithms, that defines the notion of proximity [4], parametrized by a semantic graph and a set of weighted properties.

A semantic graph represents semantic relations (arcs) between concepts (nodes). The set of weighted properties are the result of tuning the semantic relatedness algorithm in order to improve its quality. To discover which set of properties should be considered and to assign weights to each of those properties in order to maximize the algorithm's quality is the goal of this research.

The rest of paper is organized as follows. The next section describes the previous work towards a methodology for defining a semantic relatedness measure, using WordNet 2.0 as a case study. Section 3 details the challenges raised by larger semantic graphs and the strategy that is being followed to overcome them.

2. PREVIOUS WORK

This section presents the process developed for tuning weights [5] in order to optimize the semantic measure quality. This tuning process takes as input a complete semantic graph and a set of benchmark data.

The semantic measure produced by this tuning process considers *proximity* rather distance as a measure of relatedness. By definition, proximity is closeness; the state of being near as in space, time, or relationship. Rather than focusing solely on minimum path lengths, proximity balances the number of existing paths between nodes.

The main issue with this definition of proximity [4] is how to determine the weights of transitions. The naïve approach

is to assign weights based on domain knowledge. However, this approach is not based on evidence and is difficult to apply to a semantic graph with many properties. To be of practical use, the weights and the set of properties of a proximity based semantic relatedness algorithm must be automatically tuned. That tuning process relies on a genetic algorithm and on bootstrapping strategy that generates a weight assignment that produces the highest Spearman's correlation value.

These approaches were validated using WordNet 2.0 and the WordSim-353 data set. The correlation coefficient obtained was 0.42. This result is better than the best in the literature [6]. However, the validation with larger knowledge bases, such as WordNet 3.0 and Freebase, raises new issues.

3. CHALLENGES AND ONGOING WORK

Consider all the properties that connect nodes in a semantic graph. Do all the properties contribute to proximity? What if some of them spoil that proximity? Is it feasible compute proximity in large knowledge bases?

WordNet is a small semantic graph when compared to DBpedia [1] or Freebase [2]. This led to the definition of an incremental algorithm that starts with a minimal graph and enlarges it by adding properties that contribute to increase the quality of the semantic measure.

Consider a semantic graph $G = (N, A, T)$ where N is a set of nodes, A is a set of arcs, and T is a set of type of arcs. The initial graph of this incremental algorithm is $G_0 = (N, \emptyset, \emptyset)$. Each iteration builds a new graph $G_{k+1} = (N, A_{k+1}, T_{k+1})$ based on $G_k = (N, A_k, T_k)$. The new set of types T_{k+1} has all the types in T_k. In fact, several candidate G^i_k can be considered, depending on the types in $T - T_k$ that are added to T_{k+1}. The arcs of A^i_{k+1} are those in A whose type is in T^i_{k+1}. The general idea is to select the G^i_{k+1} that produces an higher increment on semantic measure quality. This algorithm stops when no candidate is able to improve it.

In general, computing the semantic measure quality of G^i_{k+1} is a time consuming task. However, there some ways to make it more efficient. It should be noticed that if G^i_{k+1} is not a connected graph then the quality measure is 0. This means that for the first iteration many G^1_{k+1} can be trivially discarded. Moreover, if $A^i_{k+1} = A^i_k$ then the semantic quality measure is the same. This insight can be used to speedup the iterative process. The paths connecting pairs of concepts using arcs in A_{k+1} are basically the same that used A_k. The new paths must appear on the nodes of previous paths and can only have arcs of types in T_{k+1}. This insight can be used to compute the quality of G^i_{k+1} incrementally based on the computation of G^i_k.

The generation of the sets T^i_{k+1} is a potential issue. Ideally T^i_{k+1} would have just one element more than T^i_k. However this may not always be possible. Consider T^i_1, the candidate sets of types for the first iteration. In most cases they will produce a disconnected graph, hence with a null semantic measure quality. They will only produce a connected graph if the selected type creates a taxonomy. In many cases this involves 2 types of arcs: one linking an instance to a class, another linking a class to its super-class. To deal with this issue the incremental algorithm attempts first to generate T^i_{k+1} such that $\sharp T^i_{k+1} = \sharp T^i_k + 1$, where $\sharp$ stands for set cardinality. In none of these improve the semantic measure quality then it attempts to generate T^i_{k+1} such that $\sharp T^i_{k+1} = \sharp T^i_k + 2$, and so forth.

Nevertheless, this incremental algorithm has a number of challenges that are currently being addressed. If the set T is very large, as is the case with Freebase, it may require an heuristic to sort properties so that the most promising are explored first. What would be that heuristic? The number of candidate graphs G^i_{k+1} may also be very large, specially if one needs to consider 2 or more types. this has also implications on the stopping condition. How can one be sure that considering a set T_{k+1} with even more types would not increase the quality measure?

4. ACKNOWLEDGEMENTS

This work is financed by the ERDF – European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT – Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project FCOMP-01-0124-FEDER-037281. Project "NORTE-07-0124-FEDER-000059" is financed by the North Portugal Regional Operational Programme (ON.2 - O Novo Norte), under the National Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF), and by national funds, through the Portuguese funding agency, Fundação para a Ciência e a Tecnologia (FCT).

5. REFERENCES

[1] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer, 2007.

[2] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. ACM, 2008.

[3] S. Harispe, S. Ranwez, S. Janaqi, and J. Montmain. Semantic measures for the comparison of units of language, concepts or instances from text and knowledge representation analysis. Technical report, Research Center, Parc scientifique G. Besse, 30035 Nîmes Cedex 1, France, 2013.

[4] J. P. Leal. Using proximity to compute semantic relatedness in rdf graphs. *Comput. Sci. Inf. Syst.*, 10(4), 2013.

[5] J. P. Leal and T. Costa. Multiscale parameter tuning of a semantic relatedness algorithm. In *SLATE*, 2014.

[6] M. Strube and S. P. Ponzetto. Wikirelate! computing semantic relatedness using wikipedia. In *AAAI*, volume 6, pages 1419–1424, 2006.