Accessing related topics through community detection in knowledge graph

Lucas Donizetti Vieira
School of Technology
University of Campinas
Limeira-SP, Brazil
l201874@dac.unicamp.br

João Roberto Bertini Junior
School of Technology
University of Campinas
Limeira-SP, Brazil
bertini@ft.unicamp.br

***Abstract*—The idea of an automated system capable of representing human language and its semantics refer to the semantic networks proposed in the early 1960s. In recent years, due to the increase in computational power and the improvement of techniques of text extraction, this concept has been considered on a large scale. Projects such as Carnegie Mellon University's Never Ending Language Learning (NELL) as a goal to learn languages from the web and, especially, in an uninterrupted way. The system learns by extracting facts from the web and inserting them into its knowledge base. The knowledge base, in turn, can be seen as a network in which vertices are instances of category and the edges represent the relations between them. This structure, referred to as knowledge graph, allows the application of methods based on complex networks for various applications in a complementary or superior way to traditional methods. This paper addresses the use of a community detection algorithm for accessing topics in a knowledge graph generated from a given topic of interest. Initial results show that the progressive division of that network, carried out by the proposed method, produces a coherent hierarchy of terms related to the topic of interest.**

***Index Terms*—Knowledge graph, community detection, topic detection, language network.**

I. Introduction

In mid-2012 Google added a new feature to its search engine and named it Knowledge Graph [1]. Although the company has coined the name, such technology has been considered by several groups in parallel [2]. These projects refer to the automatic extraction of knowledge from the web, considering various techniques and focusing on extracting knowledge in the form of facts in order to populate a knowledge base. Usually, this base is composed of categories and relations regarding two categories. A category qualifies a noun phrase, for example, *city(London)*; and, a relation establishes a connection between pairs of noun phrases such as *isCapitalOf(London, England)*. In this way, the knowledge base can be seen as a graph (or network) where instances of categories represent the vertices and the pairwise relationship between them represent the edges.

The growing interest in this type of representation is justified by the increase in computational power and by the improvements of text extraction techniques [3]. These factors have allowed considering a large-scale knowledge graph [4]. With the explosion in the size of these graphs, it is not only possible to represent several domains of knowledge in a common framework, but also to address them as a large complex network of interrelated elements. Seen in this way, methods of the complex network theory may be applied as a data mining tool, revealing hidden patterns that can be used to understand the structure and the evolution of these systems [5], [6].

In fact, according to Solé *et al.* [7], three related characteristics to complex networks [8], in particular, appear to be shared by all networks of languages. Language networks 1) are sparse, which means that the average number of connections per vertex is small; 2) have the effect of small world, that is, communication in these networks is made easier - the path between any two vertices of the network is very short; and 3) are scale-free, i.e. most of the network elements are connected to few others, while a small portion of them (the hubs) have a large number of connections.

One of the main characteristics presented in several complex networks, found in nature or constructed by man, is the presence of structures known as communities. A community is a group of vertices with many connections between them, whereas the connections between different communities are sparse [9]. The identification of communities in a network is of utmost importance for understanding the relationship between different components. For instance, it allows to reveal the hierarchical organization of vertices and to identify the function of a component based on the function of its members. Community detection has applications in many areas of science; some examples are: balancing nodes in parallel computing, circuit partitioning, telephony networks development and data clustering [10].

In this paper, we consider the application of a community detection algorithm to a knowledge graph, built from a given category, for related topics identification. Specifically, given a topic of interest, a knowledge graph is built around it with the facts and relations from the knowledge base. Then, the community detection algorithm, proposed by Newman and

Girvan [11], is applied to hierarchically divide the graph. Through this process, we hope to identify topics related to the topic of interest, establishing a hierarchy between the topic of interest and the ones found by the algorithm. This type of analysis can be useful in marketing planning, where it is desired to uncover product-related topics [12]. Also, in the designing of chatbots, to enhance the quality of the conversation, for instance, allowing the chatbot to extend the conversation by addressing a related subject [13]. And in topic modeling; topics and related topics can be retrieved from a graph built within a specific domain [14].

The remainder of the paper is organized as follows. Section II details the proposed approach for related topic identification. Section III shows the experiment results and their analysis. At last, Section IV concludes the work and draws some future directions.

II. The proposed algorithm for related topic identification

The proposed method considers building a knowledge graph around a topic of interest. Then, through an algorithm of community detection, divide the graph successively to obtain a hierarchy of related topics. Therefore, the first step of our method is building a knowledge graph from an initial category, which is the topic of interest. To build the graph we consider as a vertex those elements that either define a category, such as '*country*', and those belonging to a category, such as '*country(brazil)*'. Similarly, the edges stand either for a relation between a category and the elements belonging to it, as (country, country(brazil)) and the relations defined in the knowledge base between two categories, as *sportfansincountry(country(brazil),sport(soccer))*.

Algorithm 1 details the creation of the knowledge graph. It takes as input: a knowledge base, B, which the graph will be built from, a category representing the topic of interest, noted as c, and the maximum path size from c to any other category, noted as k. Where a path in a graph is a sequence of edges joining a sequence of distinct vertices. This last input parameter controls the size of the graph and, consequently, the extent of the search for related topics.

In the algorithm, the knowledge graph, G, is built by adding the neighbors of the input topic c to the graph, then proceeds by adding the neighbors of the neighbors, and so on, until the maximum path size from c to any other category is reached. The algorithm uses a stack, S, to build the graph, similarly to a depth-search procedure. Initially, c is pushed into the stack, and the algorithm proceeds by popping out a topic from the stack to find its neighbors. The algorithm makes a distinction between vertices representing a category and vertices representing members of a category. Each time a new category (or category member) is found, it is added to the graph. A vertex representing a category is connected to all the vertices representing members of that category. While a vertex that stands for a category member is connected to the vertex representing its category and to every other vertex representing a category member for which a relation connecting them exists. The newly found category is then pushed into S. The process ends when all the branches from c to any other vertex reaches a path of size k.

Algorithm 1: Knowledge graph construction from a topic given as input

Input: Knowledge base B, topic of interest c, maximum path size from c to any other vertex, k
Output: A knowledge graph G

```
G ← (V, E);
V ← ∅;
E ← ∅;
let S be a stack;
S.push(c);
V ← V ∪ c;
while S is not empty do
    v ← S.pop();
    if path(c, v) < k then
        if v belongs to a category h(v) then
            if h ∉ V then
                V ← V ∪ h;
                E ← E ∪ (u, h);
                S.push(h);
            end
            for all categories u in B do
                if ∃ a relation e(h(v), u) and u ∉ V
                 then
                    V ← V ∪ u;
                    E ← E ∪ (h(v), u);
                    S.push(h(u));
                end
            end
        end
        if v defines a category then
            for all members z of v, v(z), in B do
                V ← V ∪ v(z);
                E ← E ∪ (v, v(z));
                S.push(v(z));
            end
        end
    end
end
```

Once the graph has been built, the idea is to divide it into hierarchical clusters to reveal related topics. The algorithm considered for this task was the community detection algorithm proposed by Newman and Girvan [11]. Their algorithm uses the concept of betweenness as a measure of centrality of the edges. Betweenness is defined for an edge as the number of shortest paths among any pair of vertices in the graph. If a network has communities that are connected by few edges, then all the shortest paths between different communities should follow one of these few edges. Thus, the edges that connect these communities will have a high value of betweenness. By removing these edges, the groups are separated, revealing the underlying community structure

of the graph.

Algorithm 2 details the proposed method for accessing related topics. The method consists of iteratively dividing the input graph into two subgraphs. Each time a subgraph is obtained, a histogram with the categories and the number of connections they have is generated. These histograms are then used to access the prominent topic of that network. A graph is no longer considered for further division if it does not have any vertex defining a category.

Algorithm 2: Proposed algorithm for related topic analysis

Input: Graph
Output: Two graphs, and their histogram, obtained from the division of the input graph
while *The network is connected* **do**
  Calculate the betweenness for all network edges;
  Remove the leading edge of betweenness;
end
Plot histogram of subgraph G1;
if ∃ *vertices that defines a category in G1* **then**
  Call Algorithm 2 for G1
end
Plot histogram of subgraph G2;
if ∃ *vertices that defines a category in G2* **then**
  Call Algorithm 2 for G2
end

III. Experiments

For the experiments conducted in this work, we have downloaded the NELL knowledge base [15] which contains 2,211,488 categories and 2,530,397 relationships. To perform related topic analysis, two experiments were carried out, the first starting from the category 'olympics' and the second from the category 'amphibian'.

To analyze the networks, a histogram was generated relating each category with the number of vertices connected to it. For example, if the category 'sport', in a given network, has 23 connections, it will be represented as horizontal bar of size 23 in the histogram. Histograms are generated for all networks obtained by the proposed approach. From the initial network to all networks resulted from the hierarchical divisions made by the application of the algorithm.

The results were analyzed regarding the generated histograms. The more connected a category is, the higher is its importance in the network. The networks obtained and their divisions can be represented by a tree showing the hierarchy between them. In what follows, each reference to the network number in the results is related to the network number in the tree of each experiment.

A. Experiment 1

For the experiment 1, the category 'olympics' was given as input. Figures 1 and 2 correspond to the histogram representation of the initial network.

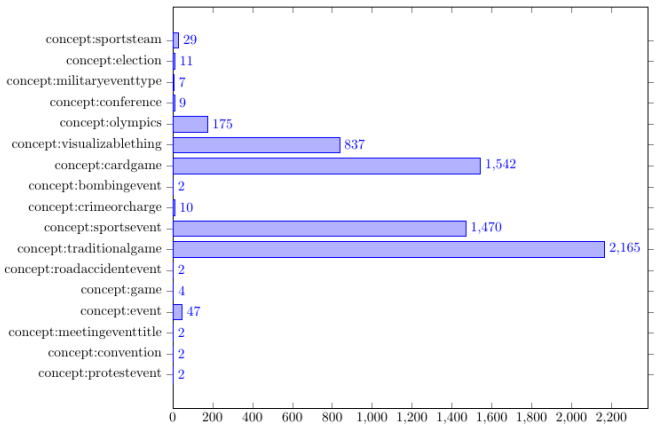

Fig. 1. Histogram of the initial network of experiment 1, part 1

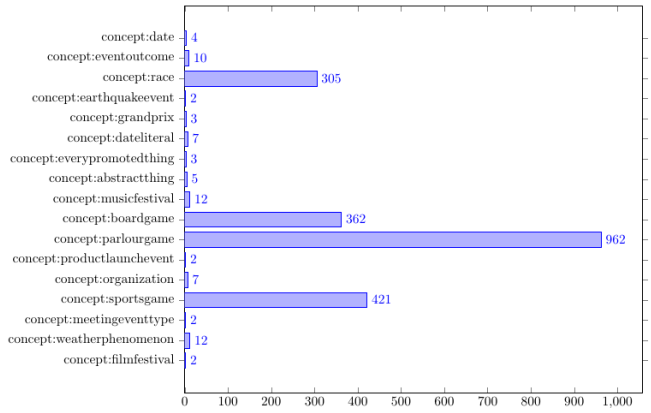

Fig. 2. Histogram of the initial network of experiment 1, part 2

The division of the initial network, yielded network 2 and network 3, whose histograms are depicted in Fig. 3 and Fig. 4, respectively. Figure 3 shows categories related to sports in general, such as card games, board games, sports games and races, whereas network 3 (Fig. 4) has a large predominance of categories related to events, such as crime or charge and weather phenomenon.

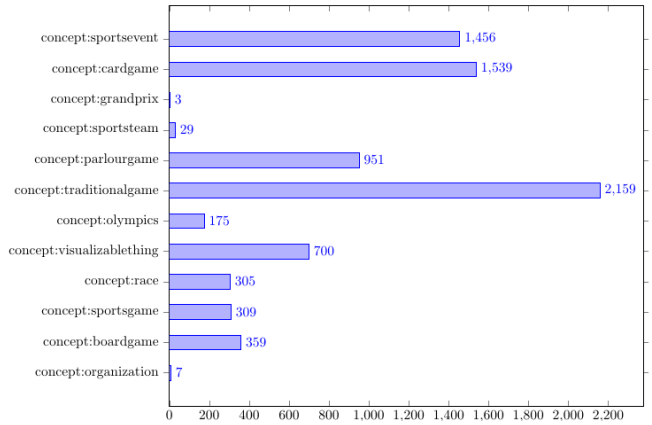

Fig. 3. Histogram of network 2 of experiment 1

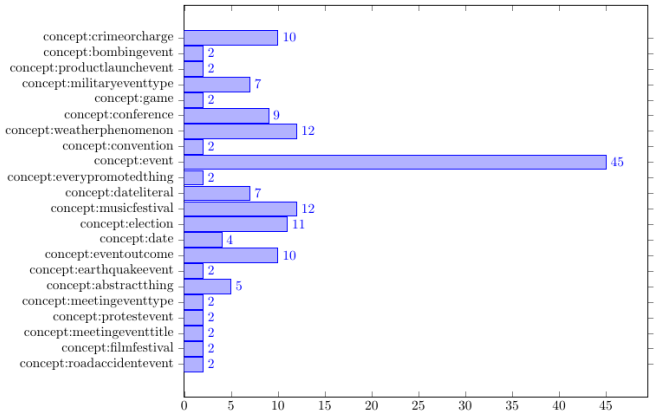

Fig. 4. Histogram of network 3 of experiment 1

Following with the division of network 3, network 6 and 7 are generated. The histogram of network 6 has categories of events related to dates. The histogram of network 7 (Fig. 5) remained with categories related to events.

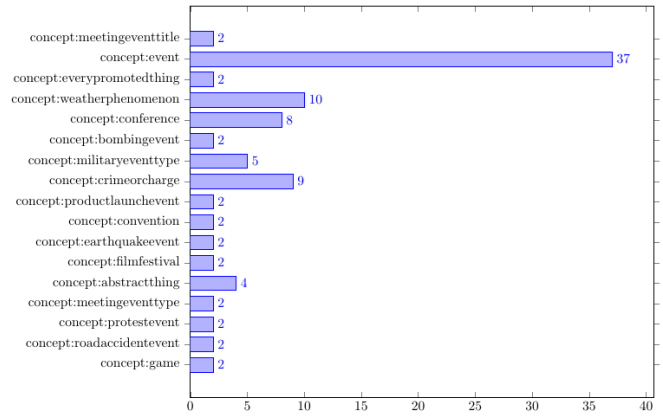

Fig. 5. Histogram of network 7 of experiment 1

The division of network 2 produced network 4 and network 5. This division has separated sports-related categories, in network 4, from organizational categories, that were grouped in network 5.

Proceeding with the division of network 4, network 8 and 9 are obtained. This division has separated the category 'games' from the rest.

The last division separates network 8 into network 10 and network 11. Network 10 ends up with categories related to traditional sports, while network 11 was left with the categories parlour games and visual categories, which are related to each other.

Figure 6 shows the hierarchy of the main topics found by the proposed approach. Each rectangle is numbered with the network it represents and it highlight the most connected topic in that network. In the left side of the tree it is possible to see that some categories are being retrieved from sports, as organizations, games and parlour games, until ending up with traditional sports. While on the right side, categories related to events have been obtained, which, like sports, they are highly related to the category 'olympics'.

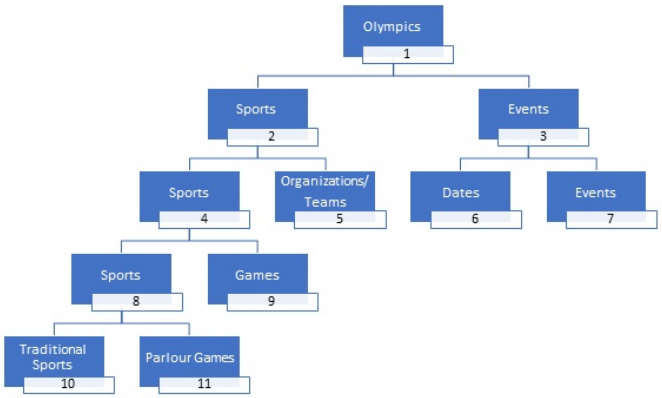

Fig. 6. Hierarchical tree of the predominance of categories resulted from experiment 1. Each rectangle represents a network and the topic on it stands for the most connected concept in that network.

B. Experiment 2

For experiment 2, the input topic was 'amphibian'. Figures 7 and 8 are the initial histogram representation of the network of this experiment.

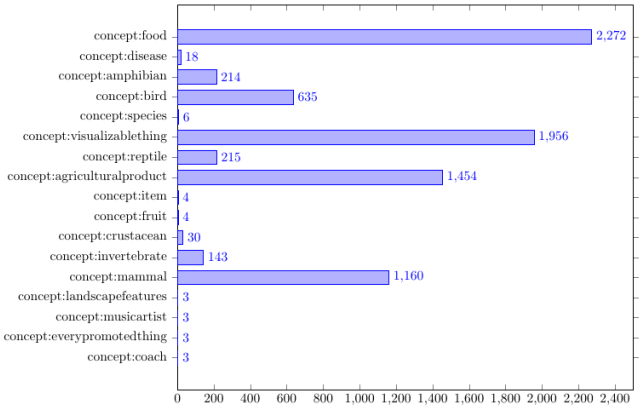

Fig. 7. Histogram of the initial network of experiment 2, part 1

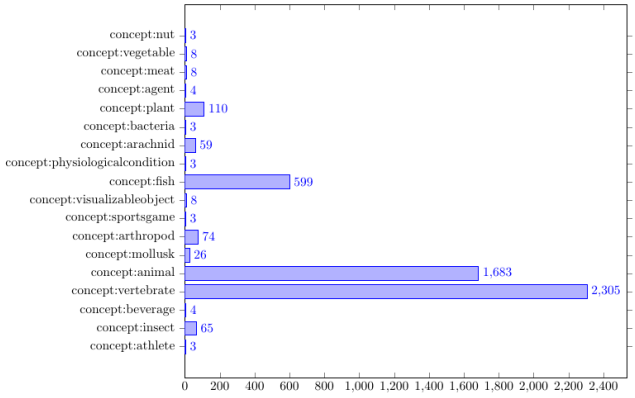

Fig. 8. Histogram of the initial network of experiment 2, part 2

The first division performed by the proposed method has separated network 1 into networks 2 and 3. Network 2 presents several concepts related to animals, whereas network 3 mostly

refers to concepts not related to animals, such as food and agricultural product.

Proceeding with the division of network 2, networks 4 and 5 has been obtained. This division has separated concepts related to living beings, in network 5 from concepts related to sports, in network 4. Sports-related concepts may have been included in the network due to the possible presence of metaphors in the knowledge base, e.g. a relation connecting 'Michael Phelps' to 'amphibian' [2]. As expected, sports and related terms, as coach, sportsgame, athlete, have formed a small community, which formed network 4.

The division of network 3 has separated only the concept representing physiological condition from species and food concepts.

The division of network 6 has further separated topics that seem unrelated to the input topic, as 'agent' and 'everypromotedthing' in network 8, from more specific, biology-related topics in network 9.

In the last division, the categories species and bacteria have been separated from network 8, to form network 11. Network 10 also resulted from the division of network 8, gathers categories related to each other, such as fruit, agricultural products and vegetables.

Figure 9 shows the hierarchy of the main topics found when the topic 'amphibian' has been given as input. In the figure it is possible to see the hierarchy of the concepts resulted from experiment 2. On the left side, as already commented, the category sports has appeared possibly due to the presence of metaphors in the text the relation was extracted from. However, in the following division the sports-related concepts have been correctly separated from living beings. On the other hand, on the right side, it can be seen that the concepts are more related to biology, such as species and physiological concepts. Also this sub-tree has highlighted concepts as food and agricultural products, that can also be traced back to the input topic. Now, the topic 'agent' is also linked to biology, for instance in sentences like 'infectious agents'[3].

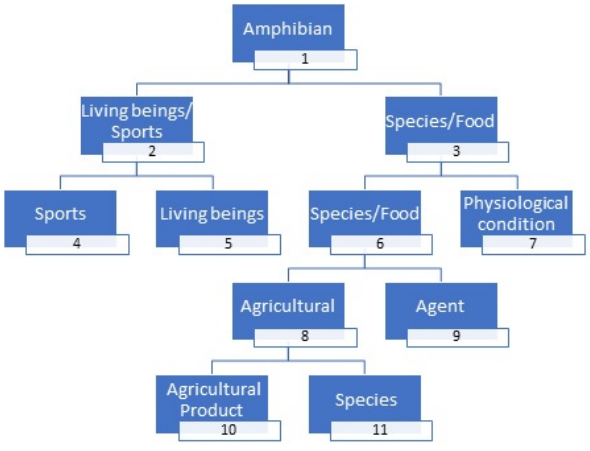

Fig. 9. Hierarchical tree of the predominance of concepts of experiment 2

[2]http://www.espn.com/espn/page2/story?page=hill/080816

[3]https://www.ncbi.nlm.nih.gov/books/NBK27114/

IV. Conclusion

This paper has proposed a new method to access related topics from a given topic of interest. The method builds a knowledge graph from a knowledge base with concepts and relations between them. Then, using a community detection algorithm, the network is iteratively divided into clusters, hierarchically exposing topics that are related to the topic of interest. Initial results show that the method is sound and may be suitable for tasks as topic modeling chatbot designing. Future work includes building a larger network around the topic of interest, applying the method to knowledge bases of specific contexts and exploring alternative ways to build and divide the graph.

References

[1] A. Singhal, "Introducing the knowledge graph: things, not strings," http://goo.gl/zivFV, 2012, accessed on 2019-10-05.

[2] H. Paulheim, "Knowledge graph refinement:a survey of approaches and evaluation methods," *Semantic Web*, vol. 8, no. 3, pp. 489–508, 2017.

[3] T. M. Mitchell, J. Betteridge, A. Carlson, E. Hruschka, and R. Wang, "Populating the semantic web by macro-reading internet text," in *Proceedings of the International Semantic Web Conference (ISWC)*, vol. 51, 2009, pp. 998–1002.

[4] J. Pujara, H. Miao, L. Getoor, and W. Cohen, "Knowledge graph identification," in *Lecture Notes in Computer Science*, vol. 8218, 2013, pp. 542—-557.

[5] S. Bornholdt and H. G. Schuster, *Handbook of Graphs and Networks: From the Genome to the Internet*. Wiley-vch, 2003.

[6] M. Zanin, D. Papo, P. Sousa, E. Menasalvas, A. Nicchi, and E. Kubik, "Combining complex networks and data mining: why and how," *Physics Reports*, vol. 635, pp. 1–44, 2016.

[7] R. V. Solé, B. Corominas-Murtra, S. Valverde, and A. L. Steels, "Language networks: Their structure,function, and evolution," *Complexity*, vol. 15, pp. 20–26, 2010.

[8] M. Newman, "The structure and function of complex networks," *SIAM Review*, vol. 45, no. 2, pp. 167–256, 2003.

[9] S. Fortunato and D. Hric, "Community detection in networks: A user guide," *Physics Reports*, vol. 659, pp. 1–44, 2016.

[10] Z. Lu, J. Wahlström, and A. Nehorai, "Community detection in complex networks via clique conductance," *Scientific Reports*, vol. 8, no. 5982, pp. 1–16, 2018.

[11] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," in *Proceedings of the National Academy of Sciences*, vol. 99, 2002, pp. 7821—-7826.

[12] J. Sterne, *Artificial Intelligence for Marketing: Practical Applications*. Wiley, 2017.

[13] P. Agarwal, M. Ramanath, and G. Shroff, "Retrieving relationships from a knowledge graph for question answering," in *Lecture Notes in Computer Science*, F. N. M. P. H. C. H. D. e. A. i. I. R. Azzopardi L., Stein B., Ed., vol. 11437. Springer, Cham, 2019, pp. 35–49.

[14] L. Yao, Y. Zhang, B. Wei, Z. Jin, R. Zhang, Y. Zhang, and Q. Chen, "Incorporating knowledge graph embeddings into topic modeling," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 2017, pp. 3119–3126.

[15] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis, T. Mohamed, N. Nakashole, E. Platanios, A. Ritter, M. Samadi, B. Settles, R. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saparov, M. Greaves, and J. Welling, "Never-ending learning," in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15)*, 2015.