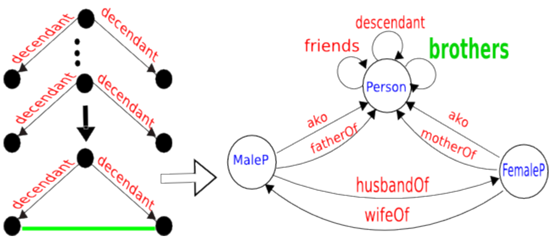

Figure 5: Example: Finding a new relation

relations among two categories to extend its ontology by mapping NELL's KB into a graph. Additionally it finds instances of the new found relations(new facts) and misplaced edges on the graph (wrong facts on NELL's OKB). Its algorithm is present in[1], it's very similar to the Relations-Predictor Algorithm, it uses a link-prediction index called extra-neighbours, and groups the open triangles in category groups either. If we uses the definition of N_o would be possible to better describe Prophet's algorithm. For curiosity purpose, currently Prophet is working with OntExt[14], prophet finds pairs of categories that might be related and OntExt finds names to these pairs.

6.2 Finding New Categories

Another disseminated topic of graph-mining the graph clustering, sometimes called community detection algorithms[6]. Briefly explaining, this task uses the topology information of the graph to divide it in clusters of nodes.

To find new categories in an N_o, we can use community detection algorithms in a similar way of the link-prediction. We apply the algorithm in the graph, then analyses each community that were found (such as the triangle category groups), and if the community attends to a given condition we can create a new category from it. Another option would be if the communities shares common features(same set of relations for example), then each community could be an instance of a new possible category (see Figure 6).

In the **Figure 6** we exemplify how we can find a new category using community detection in our imaginary social network. Imagine that we apply a community detection algorithm that returns communities where nodes inside of them are mostly related by the following relations: *descendant, fatherOf, motherOf, brothers* (Figure 6 (a)). Observing this pattern, we can infer that this communities are families, so the category "family" can be added to our N_o (Figure 6 (b)).

6.3 Finding Inference Rules (new facts)

We're currently working in a new component of NELL[4] to find inference rules through graph-mining that we call the Graph Rule Learner(GRL), it uses Prophet's link-prediction metric, the extra-neighbours index to find and make a threshold to validate the rules and choose what predicate will be the conclusion(head) of the rule.

In **line 1**, the algorithm will find and list all the closed triangles $\Delta(u, v, w)$ of the graph G, the number of neighbours $\aleph$ between each pair of nodes of the triangle are calculated (e.g $\aleph(u, v)$), and grouped in the respective open triangle category group ${\Lambda_c}^2$ (e.g $\Lambda(c_u, c_v)$). The closed triangle is also grouped in the closed triangle category group $\Delta_c(c_u, c_v, c_w)$. Next in **line 8**, for each open triangle category group $\Lambda_c(c_i, c_j)$, the number of extra neighbours $\aleph_c$ will be calculated. Then in **line 11**, for each close triangle category group $\Delta_c(c_u, c_v, c_w)$, the pair of categories with the highest extra neighbour value $\aleph_c$ will be selected, e.g (c_u, c_v). Then, if the extra neighbour value of this pair is greater or equal then a given threshold ξ, validate the rule $r_{c_u c_v}(c_u, c_v) \Longleftarrow r_{c_u c_w}(c_u, c_w) \wedge r_{c_w c_v}(c_w, c_v)$.

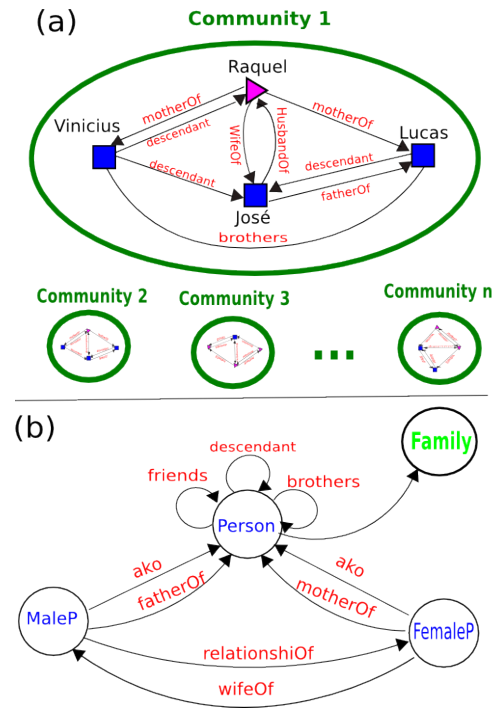

Figure 6: Example: Finding a new category

One literal $r_{c_x c_y}(c_x, c_y)$ indicates a relation(the predicate r) $r_{c_x c_y} \in E_c$ between the categories c_x and c_y, so the paremeters of the predicate $r_{c_x c_y}$ need to be instances of the categories c_x and c_y respectively.

To exemplify this in using our imaginary social network after adding the relation *brother* $< Person, Person >$, we could apply GRL and possibly obtain the rule above:

$$
\begin{aligned}
& brothers(Person_X, Person_Z) \Longleftarrow \\
& \quad descendant(Person_X, Person_Y) \\
& \quad \wedge\, descendant(Person_Z, Person_Y)
\end{aligned}
$$

This rule can be used to find more instances of the relation brother to the N_o, adding edges to G_i (the same of facts to the OKB).

After the application of the graph-mining algorithms presented in the last three subsection in our imaginary social network, in **Figure 6 (b)** the final ontological model graph is present, and in **Figure 7** we present how the ontological instances graph could be augmented with that.

7. DISCUSSIONS AND CONCLUSION

In this work we presented a structure called ontological network(N^o) that can be used to map and ontological

[2]Despite the three nodes are connected. It is considered that the edge between the pair of parameters of *Lambda* doesn't exist in each group

Algorithm 2 The GRL

Require: $G_i = (V, E, X)$
Ensure: List of Inference Rules

Find all $\Delta(u, v, w)$ in G
for all closed triangle $\Delta(u, v, w)$ **do**
  Calculate $\aleph(u, v)$, $\aleph(v, w)$ and $\aleph(w, u)$
  Group $\Delta(u, v, w)$ in $\Delta_c(c_u, c_v, c_w)$
  Group $\Lambda(u, v)$ in $\Lambda_c(c_u, c_v)$, $\Lambda(v, w)$ in $\Lambda_c(c_v, c_w)$ and $\Lambda(w, u)$ in $\Lambda_c(c_w, c_u)$
end for
for all $\Lambda_c(c_i, c_j)$ **do**
  Calculate

$$\aleph_c(c_i, c_j) = \sum_{\forall \Lambda(u,v) \in \Lambda_c(c_1, c_2)} (\aleph(u, v) - 1)$$

end for
for all $\Delta_c(c_u, c_v, c_w)$ **do**
  Find the category pair with highest $\aleph_c$: $(c_i, c_j) = MAX(\aleph_c(c_u, c_v), \aleph_c(c_v, c_w), \aleph_c(c_w, c_u))$
  if $\aleph_c(c_i, c_j) \geq \xi$ **then**
    Validate the rule: $r_{c_i c_j}(c_i, c_j) \Longleftarrow r_{c_i c_k}(c_i, c_k) \wedge r_{c_k c_j}(c_k, c_j)$
  end if
end for

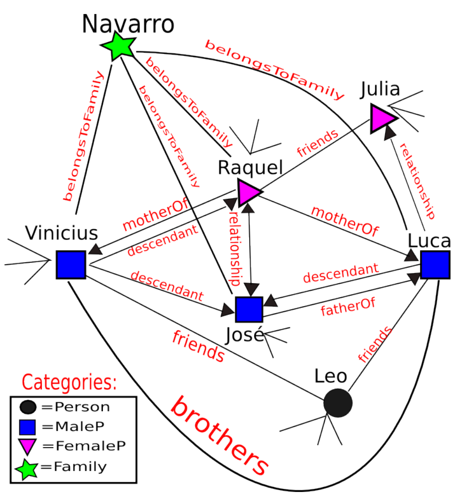

Figure 7: New Ontological Instances Graph

knowledge base(OKB) for the execution of graph-mining algorithms to extract implicit information from it. The goal we wanted to achieve with this article(and the N^o structure) is to formally define a structure to be used assisting another projects that uses ontological knowledge bases and intends to map it into graphs to apply graph-mining algorithms on it. We also wanted to present some ideas and demonstrate by using simple algorithms that graph mining techniques can be very useful to extends OKBs.

We already implemented a version of Prophet within OntExt and the Graph Rule Learner, both functional to work with NELL. We also start to work on the task of find new categories. The structure (N^o) was widely used in the design and documentation of all these algorithms. This project still lack of proof of how this structure is really usefull and measure it's usefullness and other characteristics of it, but we plan to work on that in a near future.

8. REFERENCES

[1] A. P. Appel and E. R. Hruschka Junior. Prophet – a link-predictor to learn new rules on nell. In *Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops*, ICDMW '11, pages 917–924, Washington, DC, USA, 2011. IEEE Computer Society.

[2] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Pate-Schneider. *The description logic handbook: Theory, implementation, and applications.* Cambridge University Press, 2 edition edition, 2010.

[3] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *In Proceedings of SIGMOD*, 2008.

[4] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. H. Jr., and T. M. Mitchell. Toward an architecture for never-ending language learning. In *Proceedings of AAAI*, 2010.

[5] I. L. Dong, K. Murphy, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, T. Strohmann, S. Sun, and W. Zhang. Knowledge vault: A web-scale approach to probabilistic knowledge fusion, 2014.

[6] S. E. S. Graph clustering. *Computer Science Review*, 1(1):27–64, 2007.

[7] L. Getoor and C. P. Diehl. Link mining: a survey. *ACM SIGKDD Explorations Newsletter*, 7:3–12, 2005.

[8] S. Kelley, M. Goldberg, M. Magdon-Ismail, K. Mertsalov, W. Wallace, and M. Zaki. graphont: An ontology based library for conversion from semantic graphs to jung. *Intelligence and Security Informatics, ISI'09*, pages 170–172, 2009.

[9] A. Lancichinetti and S. Fortunato. Community detection algorithms: A comparative analysis. *Physical Review E*, 80(5), 2009.

[10] N. Lao, T. Mitchell, and W. W. Cohen. Random walk inference and learning in a large scale knowledge base. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 529–539, Edinburgh, Scotland, UK., July 2011. Association for Computational Linguistics.

[11] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *Proceedings of CIKM*, pages 556–559, New York, NY, USA, 2003. ACM.

[12] F. Lorrain and H. White. Structural equivalence of individuals in social networks. *Journal of Mathematical Sociology*, 1:49–80, 1971.

[13] S. Milgram. Tje small-world problem. *Psychology Today*, pages 62–67, 1967.

[14] T. Mohammed, E. R. Hruschka Jr., and T. M. Mitchell. Discovering relations between noun categories. *EMNLP*, ACL:1447–1455, 2011.

[15] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: a core of semantic knowledge. In *In Proceedings of WWW*, 2007.

[16] P. Velardi, S. Faralli, and R. Navigli. Ontolearn reloaded: A graph-based algorithm for taxonomy induction. *Computational Linguistics Journal*, 2013.

[17] D. J. Watts and S. H. Strogatz. Collective dynamics of small-world networks. *Nature*, 393(6684):440–442, June 1998.