Explicative Path Finding in a Semantic Network

Kévin Cousot, Mathieu Lafourcade
LIRMM / Campus St Priest, 161 Rue Ada, 34090 Montpellier, France
kevin.cousot@lirmm.fr, mathieu.lafourcade@lirmm.fr

Introduction

When building a knowledge base (KB), it is desirable to be able to assess its quality, and one approach to undertake such a task is to exhibit explanation(s) related to contained information. For example, from an information stating that (a) *tiger 'characteristic' dangerous*, we would like to be able to ask *Why ?* and obtaining at least one explanation, for example *tiger 'is-a' 'wild animal' characteristic' dangerous* or something more elaborated like *tiger 'agent' attack 'consequence' death & dangerous 'consequence' death*. These explanations take the form of *explicative* path in the KB, and can be good clues of the quality of the knowledge present in the database. To do so, the KB has to be represented (or converted) under the form of a lexico-semantic network (for instance, like Wordnet or Babelnet).

After presenting some related work, we detail some aspects of the JeuxDeMots lexical network on which we undertook our modeling and experiment. Then, we introduce our method, the *explicative path finding by triangulation* and some evaluation.

1 Related work

Path finding in semantic resources is used in several applications. Path finding has been used in *information retrieval* to improve ranking over semantic web resources and results relevance (Lee et al., 2009). In that work, paths are explored in an ontology from user query keywords and available resources. Every path is not meaningful and, in order to limit processing useless paths only those matching some practical constraints are considered. Typically, for a path to be valid it is required that a relation's range is the same as the next relation's domain (or a subclass of it). Thus, paths look like : $writtenBy^{-1}(Prof, Publication) \wedge hasTitle(Publication, Str)$. The notation with the R^{-1} is R with the order of its arguments inverted. Once meaningful paths are found, they are then weighted according to the information content of their relations and mutual information of the entities they link. With those metrics it is possible to order paths according to their relevance. Path length is taken into account by decreasing the path weight as length increases because shorter paths are assumed to be more relevant. Finally, retrieved resources are ranked according to the number of meaningful paths, keyword coverage and their distinguishability.

In *text-mining*, (Song et al., 2015) use semantic paths to explore large text corpora and try to find relations between entities such as treatments and secondary effects. To perform closed-discovery, two terms are given to the system, source and target, which tries to infer intermediate terms between them. Intermediate, or linking terms are obtained from text corpora using text mining techniques. In particular, entities are extracted using a NER software and mapped to Unified Medical Language System (UMLS) while relations come directly from biomedical verbs identified in the corpora. The same process is applied to the linked terms, building a graph step after step. The graph's edges are weighted according to several similarity measures such as Path (Lin, 1998) and LCH (Leacock and Chodorow, 1998). Eventually, found paths are evaluated by experts (people in NLP) and native speakers.

Some research focused on causality. In (Besnard et al., 2008) an inference system able to provide explanations of causal statements is developed. Based on first order logic, it introduces a language to express cau-

sal statements (such as *On(alarm) causes Heard(bell)*) and truths (*Heard(soft_bell)* $\Longrightarrow$ $\neg$ *Heard(loud_bell)*). In this language, predicates are unary and express facts related to entities whereas constants are elements of an ontology listing (entities and their is-a relationships). From this, causal (α causes β) and explanation (α explains β because ϕ) atoms are introduced. A set of patterns manually defined then make it possible to infer explanations by exploiting the ontology (which allows generalization : if b is-a a then $p(b)$ entails $p(a)$) and explanation transitivity.

2 RezoJDM : the JeuxDeMots lexical-semantic network

JeuxDeMots (Lafourcade, 2007) is a construction environment for RezoJDM, a lexical-semantic network for French. It is a graph whose vertices are labeled with terms, concepts or any kind of text expression. Edges are oriented, weighted and typed with either lexical (lemma, location, action to verb, ...) or semantic (hypernymy, meronymy, agent, ...) relations. Over 100 different lexical and semantic types are available. In the following, we use indistinctly the terms *edge* and *relation*. In particular, polysemy, or some more precise usage of a word is expressed through the semantic refinement (*raff_sem*) relation type. For example, the word *avocat* has two meanings : it is either referring to *lawyer* (*juriste*) or *avocado* (*fruit*). This knowledge is encoded in the network with 3 vertices and 2 edges :

$$avocat \xrightarrow{raff\_sem} avocat{>}juriste$$
$$avocat \xrightarrow{raff\_sem} avocat{>}fruit$$

The weight of the relation expresses the strength o f association with the following principle : the higher the weight, the more relevant the relation between the terms (relatively to other relations with lower weight). Impossibilities and exceptions are identified with a negative weight. Typically : *fly* $\xrightarrow{\text{agent/-100}}$ *ostrich*. The weights are the result of the player activity in the JeuxDeMots games, i. e. the more a term is associated by player the highest the relation weight. Labelled vertices can also be linked to miscellaneous informations such as their polarity (positive, negative, neutral), some conceptual information, their color if any, or even a political connotation.

RezoJDM is built by combining different inputs. For the most part, data is collected via GWAPs [1] in which players participate to the network expansion by providing new terms and relations or consolidating them. Direct contributions can be done through Diko [2] a collaborative dictionary that allows users to edit the network and add, validate or correct knowledge data. Different inference mechanisms also continuously explore the network, generating new data (Zarrouk et al., 2013). So far, RezoJDM has more than 1.5 millions vertices and 100 millions relations.

3 Explicative path finding by triangulation

RezoJDM is a small world network : the diameter is quite small (around 6) as terms are often linked to hubs (terms with a large number of edges, such as *animal*, *person*, *place*, *process*, etc. The weights of relations are distributed according a power law. Theses properties are very interesting as it makes many exploration heuristics possible and as such a lot of raw facts are easily accessible through high-level knowledge. For example, the following fact is correct :

$$drug \xrightarrow{against} disease$$

But this is not self-explanatory and it does not tell us very well why drugs are acting against diseases.. We believe more information would be inferred by focusing on paths instead on direct relations :

$$drug \xrightarrow{instr^{-1}} healing \xrightarrow{against} disease$$

The interest of an explicative path is to provide such an explanation, which gives clues of the rightfulness of the KB. Wrong paths can pinpoint defects in the KB.

3.1 Our Approach

The inference of explanatory paths is based on **triangulation** : it is a matter of completing triangular relations arrangements by using the KB to find the relationship (s) that are lacking for the triangle to be complete.

1. Game With A Purpose
2. http ://www.jeuxdemots.org/diko.php

Starting from a true fact $x \xrightarrow{t} y$, **induction** consists in finding an intermediary vertex v connecting x and y. In order to maintain consistency, the (v, y) edge must have the same relation type t as the starting fact while (x, v) can be of any type. Not considering this constraint definitively leads to an increased number of nonsensical inferences. The result is an explicative path of length 2 (Figure 1).

For instance, to explain why *tiger* $\xrightarrow{carac}$ *dangerous*). we would like to get *tiger* $\xrightarrow{is\text{-}a}$ *wild animal* $\xrightarrow{carac}$ *dangerous*.

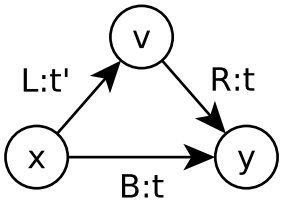

FIGURE 1 – Triangle schema with L(eft), (R)ight and B(ase) relations. Induction proposes possible pairs (LR) of relations t and t' from the initial (B) relation t. Conversely, consolidation proposes a possible initial relation (B) from a pair (LR).

The process is then applied recursively to the path's edges, expanding it and refining the explanation.

Beside the *triangulation*, there is dual operation named **consolidation**, which aims at finding in the KB a shortcut between two connected edges. In (Figure 1) finding a relation t from x, y is a consolidation. The consolidation task is undertaken along with induction.

The intermediary vertex choice is critical and one must choose a path that provides useful information. To tackle this task we use the confidence index from the data mining domain. Given two events X and Y, the confidence in the rule $X \rightarrow Y$ (X gives Y) is the quotient of X and Y joint probability and the probability of X :

$$conf(X \Rightarrow Y) = \frac{P(X,Y)}{P(X)} = \frac{P(RB)}{P(L)}$$

In our case, X is the number of occurrences of the following triangle's pattern (right hand side and base, RB) : $x \xrightarrow{t} y \xleftarrow{t} v$, while Y is the triangle's left hand side (L) : $x \xrightarrow{t'} v$. When co-occurring, X and Y form the triangle. Because only relation types (and not vertices) are relevant, we write $conf(t, t')$. For example, if the network contains 131 810 RB typed *agent* and 24 005 triangles with *instr* types for L then $conf(agent, instr) = 0.182$. At each step, the 3 vertices with the highest confidence are chosen. We also add an arbitrary threshold under which vertices are ignored. Other measures could have been considered such as activation distance (Lafourcade, 2011) or Pointwise Mutual Information (Bouma, 2009).

One question arises as why not is the confidence $conf(X \Rightarrow Y)$ defined as $\frac{P(RLB)}{P(B)}$ (instead of $\frac{P(RB)}{P(L)}$) ? This comes from the two constraints we put on ourselves : first, we want triangles, second, B and R must be of same type. Therefore the confidence measure must allow us to chose the best amongst all possible L of type t'.

The confidence measure allows us to choose some intermediary vertices, nevertheless we also need a way to limit the graph's exploration. It is indeed pointless to continue the path's expansion to exhaustion (and certainly not computationally sound). As the path gets longer, it becomes more likely that it will contain irrelevant informations, we therefore limit its length. Likewise we limit the distance between the intermediary vertex and the starting fact vertices by using Jaccard index (Jaccard, 1901) :

$$J(x, y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|}$$

with x and y being two vertices and Γ the neighbor function which is the proportion of common neighbors. If a given J value falls below a given threshold, the expansion is canceled.

4 Evaluation

For the evaluation, the following parameters were used. We used the following semantic relation types : *is-a, charac(teristic), (has-)parts, place, against, agent, patient, instrument, consequence, implication and raff_sem.* Explicative paths length is limited to 5. We also set the minimum J to 1/3 and the minimum confidence to 0.5. We undertook two experiments and the produced explicative paths have been manually evaluated :

. On the medicine domain (M1) : 100 facts (related to this domain) have been randomly selected in the KB ;
. On general common sense (CS1) : 100 facts have been hand chosen (like *kettle* $\xrightarrow{agent}$ *burn*).

4.1 Results

Results are presented respectively in tables 1 and 2. Each path has been evaluated as valid, not valid or borderline (used when the evaluators have some difficulties evaluating the soundness of the path). Some acceptable path examples : *kettle* $\xrightarrow{place}$ *fire* $\xrightarrow{charac}$ *hot* $\xrightarrow{conseq}$ *burn*; and *avocado* $\xrightarrow{is\text{-}a}$ *fruit* $\xrightarrow{parts}$ *rind* $\xrightarrow{patient\text{-}of}$ *peel*;
Some borderline paths : *avocat* $\xrightarrow{is\text{-}a}$ *homme* $\xrightarrow{parts}$ *peau* $\xrightarrow{patient\text{-}of}$ *peler*; (Eng. *avocado/lawyer* $\xrightarrow{is\text{-}a}$ *man* $\xrightarrow{parts}$ *skin* $\xrightarrow{patient\text{-}of}$ *to peal*;) (as stated before, *avocat* and has two meanings : *lawyer* and *avocado*).

l	nb	valid	not valid	borderline
1	130	89	5	6
2	259	87	6	7
3	345	80	9	11
4	307	76	12	12
5	167	69	16	15

TABLE 1 – M1 results in % according to length of path (l). nb is the number of paths.

l	nb	valid	not valid	borderline
1	176	83	7	10
2	312	79	10	11
3	406	74	12	14
4	367	73	12	15
5	216	67	16	17

TABLE 2 – CS1 results in % according to length of path (l). nb is the number of paths.

The agreement between validators was fairly high, around 0.78 (the percentage of the common evaluation). They were arguing mostly on borderline paths which acceptability might vary accordingly with the validator. There was up to 5 validators and they were able to discuss their choice remotely through chat systems (Skype, Hangout, etc.). Validators were people involved somehow in the JeuxDeMots project and all of them have university degrees.

4.2 Discussion

First, we can compare the results for both experiments. CS1 seems more productive (as regards the number of paths) : from the same number of initial facts (100), CS1 systematically produces more path than M1. We should keep in mind than the number of paths in Table 1 and 1 does not represent all possible paths but only those selected by our method.

The method does not produce as good results on CS1 than M1, as there is always a higher percentage of invalid or borderline paths. A beginning of explanation might be that the common language is certainly more polysemous and vague than that of a specific domain like medicine. Another possible reason would be the judgment of the evaluators. They are certainly stricter in the general domain (common sens) which is often more meaningful for them than a specific domain.

In both experiments, the number of paths produced increases with length up to 3, then decreases. The negative impact of length tends to strongly filter the number of paths, hence there is not a combinatorial explosion of paths.

Conclusion

In this article we have presented an approach for computing explicative paths in a lexical-semantic network. We undertook some experiments through the JeuxDeMots network. The preliminary results we detailed are quite encouraging as they effectively allowed to assess the network quality and consolidate many knowledge tidbits.

As future work, we aim at comparing and combining path and to do such some similarity functions should be defined on paths of different length. Furthermore, identifying almost complete paths seems to be a good direction for a new type of inference, where some extended context could be taken into account.

References

Besnard, P., M.-O. Cordier, and Y. Moinard (2008). Ontology-based inference for causal explanation.

Bouma, G. (2009). Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*.

Jaccard, P. (1901). *Etude comparative de la distribution florale dans une portion des Alpes et du Jura*. Impr. Corbaz.

Lafourcade, M. (2007). Making people play for Lexical Acquisition with the JeuxDeMots prototype. In *7th International Symposium on Natural Language Processing (SNLP'07)*.

Lafourcade, M. (2011). *Lexique et analyse sémantique de textes - structures, acquisitions, calculs, et jeux de mots*. Ph. D. thesis.

Leacock, C. and M. Chodorow (1998). Combining local context with WordNet similarity for word sense identification.

Lee, J., J. Min, and C. Chung (2009). An Effective Semantic Search Technique Using Ontology. In *Proceedings of the 18th International Conference on World Wide Web*.

Lin, D. (1998). An Information-Theoretic Definition of Similarity.

Song, M., G. E. Heo, and Y. Ding (2015). SemPathFinder : Semantic path analysis for discovering publicly unknown knowledge. *Journal of Informetrics*.

Zarrouk, M., M. Lafourcade, and A. Joubert (2013). Inference and Reconciliation in a Crowdsourced Lexical-Semantic Network. *Proceedings of the 14th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*.