From Generalization of Syntactic Parse Trees to Conceptual Graphs

Boris A. Galitsky[1], Gábor Dobrocsi[2], Josep Lluis de la Rosa[1]
and Sergey O. Kuznetsov[3]

[1] Univ. Girona Spain
[2] Univ Miskolc Miskolc Hungary
[3] Higher School of Economics, Moscow Russia

Abstract. We define sentence generalization and generalization diagrams as a special sort of conceptual graphs which can be constructed automatically from syntactic parse trees and support semantic classification task. Similarity measure between syntactic parse trees is developed as a generalization operation on the lists of sub-trees of these trees. The diagrams are representation of mapping between the syntactic generalization level and semantic generalization level (anti-unification of logic forms). Generalization diagrams are intended to be more accurate semantic representation than conventional conceptual graphs for individual sentences because only syntactic commonalities are represented at semantic level.

1 Introduction

Proceeding from parsing to semantic level is an important task toward natural language understanding, and has immediate applications in tasks such information extraction and question answering [1,4,7]. In the last ten years there has been a dramatic shift in computational linguistics from manually constructing grammars and knowledge bases to partially or totally automating this process by using statistical learning methods trained on large annotated or non-annotated natural language corpora.

In this study we attempt to approach conceptual tree using pure syntactic information such as syntactic parse trees. We explore the possibility of high-level *semantic* classification of natural language sentences based on *full syntactic parse trees*. We address semantic classes which appear in information extraction and knowledge integration problems usually requiring deep natural language understanding [2,3,5]. One of such problems is search relevancy, measuring semantic similarity between questions and answers by matching respective parse trees.

The main question of this study is what kind of semantic patterns can be inferred from complete parse tree structure. We believe that applying graph-based machine learning technique to such structure as syntactic trees, which have rather weak links to high-level semantic properties, can deliver satisfactory semantic classification results.

Learning syntactic parse trees allows performing semantic inference in a domain-independent manner without using ontologies. We apply parse tree generalization techniques to solving the following the problem of classifying search results in respect to being relevant and irrelevant to search query.

M. Croitoru, S. Ferré, and D. Lukose (Eds.): ICCS 2010, LNAI 6208, pp. 185–190, 2010.

2 Generalizing Natural Language Sentences

To measure similarity of abstract entities expressed by logic formulas, a least-general generalization was proposed for a number of machine learning approaches, including explanation based learning and inductive logic programming. It is the opposite of most general unification [6] therefore it is also called anti-unification. To measure similarity between natural language (NL) expressions, we extend the notion of generalization from logic formulas to syntactic parse trees of these expressions. If it were possible to define similarity between natural language expressions at pure semantic level, least general generalization would be sufficient. However, in horizontal search domains where construction of full ontologies for complete translation from NL to logic language is not plausible, therefore extension of the abstract operation of generalization to syntactic level is required. Rather then extracting common keywords, generalization operation produces a syntactic expression that can be semantically interpreted as a common meaning shared by two sentences.

1) Obtain parsing tree for each sentence. For each word (tree node) we have lemma, part of speech and form of word information. This information is contained in the node label. We also have an arc to the other node.
2) Split sentences into sub-trees which are phrases for each type: verb, noun, prepositional and others; these sub-trees are overlapping. The sub-trees are coded so that information about occurrence in the full tree is retained.
3) All sub-trees are grouped by phrase types.
4) Extending the list of phrases by adding equivalence transformations. Generalize each pair of sub-trees for both sentences for each phrase type.
5) For each pair of sub-trees yield an alignment, and then generalize each node for this alignment. For the obtained set of trees (generalization results), calculate the score.
6) For each pair of sub-trees for phrases, select the set of generalizations with highest score (least general).
7) Form the sets of generalizations for each phrase types whose elements are sets of generalizations for this type.
8) Filtering the list of generalization results: for the list of generalization for each phrase type, exclude more general elements from lists of generalization for given pair of phrases.

For a pair of phrases, generalization includes all *maximum* ordered sets of generalization nodes for words in phrases so that the order of words is retained. In the following example

To buy digital camera today, on Monday

Digital camera was a good buy today, first Monday of the month

Generalization contains {*digital - camera , today – Monday*} , where part of speech information is not shown. *buy* is excluded from both generalizations because it occurs in a different order in the above phrases. *Buy - digital - camera* is not a generalization because *buy* occurs in different sequence with the other generalization nodes.

Result of generalization can be further generalized with other parse trees or generalizations. For a set of sentences, the totality of generalizations forms a lattice: order on generalizations is set by the subsumption relation and generalization score. Generalization of parse trees obeys the associativity by means of computation: it has to be verified and resultant list extended each time new sentence is added.

3 From Generalization to Logical Form Representation

We now demonstrate how the generalization framework can be combined with the semantic representation such as logic forms to perform learning of text meaning. We have demonstrated how semantic features can be deduced from syntactic parse trees when appropriate similarity operation is found. However in a number of applications certain semantic knowledge is available, so it does not have to be learned. In this section we show how to combine pre-set semantic information with the learned one to build most accurate semantic representation.

We use notes from a number of customers of a bank. The dataset of three paragraphs is introduced:

1p. A friend transferred funds from a checking to a savings account. He then used the saving funds to pay for his mortgage.
2p. Premier account customers decided to transfer their funds from premier to regular savings account. The couple then used their premier account for automated mortgage payment.
3p. A mortgagee customer transferred the mortgage account from fixed to adjustable. She then decided to use the remaining funds as a last payment of mortgage for her second home.

To demonstrate a deep level understanding of meanings of these paragraphs, let us introduce two classes of “individual bank users” and “corporate bank users” and demonstrate how these classes can be formed from our data and classification performed. Notice that there is no explicit indication of belonging to one of this classes, it has to be inferred from text. There could be other classes where semantic information has to be inferred such as ‘obtained funds are used for something’ and ‘no such statement is made’, ‘account type transfer’ and ‘refinancing’, and many more.

We intend to express commonalities between the elements of training set to ‘explain’ belonging to a class, following the classical methodology of induction (Mill 1843). We hypothesize that common linguistic features of a training set *cause* the target feature (the class). In this section we form these features on both syntactic level by means of generalization and on semantic level by means of logical anti-unification. To do that, we will first proceed on syntactic level, and then show how it can be done on semantic level of logic forms. Then we finally show how the syntactic level can be mapped into semantic one. Single lines depict generalizations for the first sentence of each paragraph (a), double line – for the second sentence (b). There are multiple sentences appearing in different order in a general case. The lattice depicts the relation of “being more general’ between generalization results.

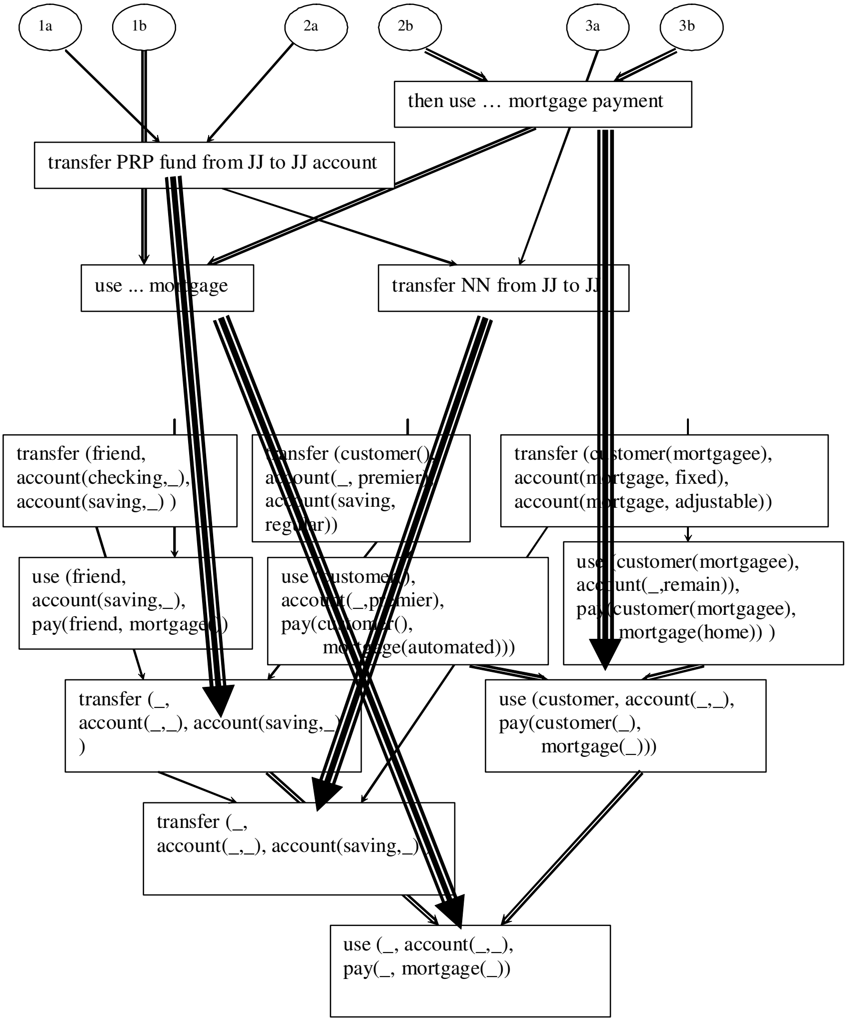
1a
1b
2a
2b
3a
3b
then use … mortgage payment
transfer PRP fund from JJ to JJ account
use ... mortgage
transfer NN from JJ to JJ
transfer (friend, account(checking,_), account(saving,_))
transfer (customer(), account(_, premier), account(saving, regular))
transfer (customer(mortgagee), account(mortgage, fixed), account(mortgage, adjustable))
use (friend, account(saving,_), pay(friend, mortgage()))
use (customer(), account(_,premier), pay(customer(), mortgage(automated)))
use (customer(mortgagee), account(_,remain)), pay(customer(mortgagee), mortgage(home)))
transfer (_, account(_,_), account(saving,_))
use (customer, account(_,_), pay(customer(_), mortgage(_)))
transfer (_, account(_,_), account(saving,_))
use (_, account(_,_), pay(_, mortgage(_))

Fig. 1. Conceptual graph for tree paragraphs

To define mapping into logic forms, we need to form logical predicates and specify semantic types of their arguments. For a pair sentences, we can first generalize them and then translate result into a logic form. Alternatively, we can translate each sentence into logic form first and then anti-unify these logic forms. Fig. 1 shows multiple paths to the results of operations of generalization and anti-unification. There is a criterion for optimal path: the resultant score of expression. For a logic form, the score is a number of terms in the expression; this fits well the score of generalization. We define an *optimal path* to the logic form of a set of samples as the one leading to the resultant logic form with the highest score.

In Fig. 1 we visualize our version of the conceptual graph for three sentences above. There is a lattice of generalizations for three paragraphs from positive set (on the top), and a lattice of anti-unifications for three paragraphs (on the bottom). There is a mapping between syntactic and semantic levels. Results of generalization of sentences are mapped into anti-unification of respective logic forms.

4 Evaluation and Conclusion

Evaluation of search included an assessment of classification accuracy for search results as relevant and irrelevant. Since we used the generalization score between the query and each hit snapshot, we drew a threshold of five highest score results as relevant class and the rest of search results as irrelevant. We used the Yahoo search API and applied the generalization score to find the highest score hits from first fifty Yahoo search results. We then consider the first five hits with the highest generalization score (not Yahoo score) to belong to the class of relevant answers. Third and second rows from the bottom contain classification results for the queries of 3-4 keywords which is slightly more complex than an average one (3 keywords); and significantly more complex queries of 5-7 keywords respectively.

The total average accuracy (F-measure) for all above problems is 79.2%. Since the syntactic generalization was the only source of classification, we believe the accuracy is satisfactory. A practical application would usually use a hybrid approach with rules and keyword statistic which would deliver higher overall accuracy, but such application is beyond the scope current paper. Since the generalization algorithm is deterministic, higher accuracy can be also achieved by extending training set.

Table 1. evaluation of classification accuracy

Type of search query	Relevancy of Yahoo search, %, averaging over 10	Relevancy of re-sorting by generalization, %, averaging over 10	Relevancy comp to baseline, %
3-4 word phrases	77	77	100.0%
5-7 word phrases	79	78	98.7%
8-10 word single sentences	77	80	103.9%
2 sentences, >8 words total	77	83	107.8%
3sentences,>12 words total	75	82	109.3%

In this study we demonstrated that such high-level sentences semantic features as *being informative* can be learned from the low level linguistic data of complete parse tree. Unlike the traditional approaches to *multilevel* derivation of semantics from syntax, we explored the possibility of linking low level but detailed syntactic level with high-level pragmatic and semantic levels *directly*.

For a few decades, most approaches to NL semantics relied on mapping to First Order Logic representations with a general prover and without using acquired rich knowledge sources. Significant development in NLP, specifically the ability to acquire knowledge and induce some level of abstract representation is expected to support more sophisticated and robust approaches. A number of recent approaches are based on shallow representations of the text that capture lexico-syntactic relations based on dependency structures and are mostly built from grammatical functions extending keyword matching (Durme et al 2003). Similarly to the above studies, we address the semantic inference in a domain-independent manner. Syntactic match allows solving problems of semantic relevancy without use of ontologies, therefore finding a number of commercial applications including relevancy engine at citizens' journalism portal AllVoices.com.

References

1. Allen, J.F.: Natural Language Understanding. Benjamin Cummings (1987)
2. Banko, M., Cafarella, J., Soderland, S., Broadhead, M., Etzioni, O.: Open information extraction from the web. In: Proceedings of the Twentieth International Joint Conference on Artificial Intelligence, Hyderabad, India, pp. 2670–2676. AAAI Press, Menlo Park (2007)
3. Dzikovska, M., Swift, M., Allen, J., de Beaumont, W.: Generic parsing for multi-domain semantic interpretation. In: International Workshop on Parsing Technologies (Iwpt 2005), Vancouver BC (2005)
4. Cardie, C., Mooney, R.J.: Machine Learning and Natural Language. Machine Learning 1(5) (1999)
5. Galitsky, B.: Natural Language Question Answering System: Technique of Semantic Headers. Advanced Knowledge International, Australia (2003)
6. Robinson, J.A.: A machine-oriented logic based on the resolution principle. Journal of the Association for Computing Machinery 12, 23–41 (1965)
7. Ravichandran, D., Hovy, E.: Learning surface text patterns for a Question Answering system. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002), Philadelphia, PA (2002)
8. Durme, B.V., Huang, Y., Kupsc, A., Nyberg, E.: Towards light semantic processing for question answering. In: HLT Workshop on Text Meaning (2003)