Kernel-based Logical and Relational Learning with kLog for Hedge Cue Detection

Mathias Verbeke[1], Paolo Frasconi[2], Vincent Van Asch[3], Roser Morante[3], Walter Daelemans[3], and Luc De Raedt[1]

[1] Department of Computer Science, Katholieke Universiteit Leuven, Belgium
[2] Dipartimento di Sistemi e Informatica, Università degli Studi di Firenze
[3] Department of Linguistics, Universiteit Antwerpen, Belgium

Abstract. Hedge cue detection is a Natural Language Processing (NLP) task that consists of determining whether sentences contain unreliable or uncertain information. This binary classification problem, i.e. distinguishing factual versus uncertain sentences, only recently received attention in the NLP community. We use kLog, a new logical and relational language for kernel-based learning, to tackle this problem. We present results on the CoNLL 2010 benchmark dataset that consists of a set of paragraphs from Wikipedia, one of the domains in which uncertainty detection has become important. Our approach shows competitive results compared to state-of-the-art systems.

1 Introduction

Information Extraction (IE) is a subdomain of Natural Language Processing (NLP) concerned with the automatic extraction of structured, factual information from unstructured or semi-structured machine-readable texts. Since it has been shown that a number of IE tasks, such as question answering systems [3] and IE from biomedical texts [4, 5], benefit from being able to distinguish facts from unreliable or uncertain information, research about hedge cue detection has increased in recent years.

Hedge cues are linguistic devices that indicate whether information is being presented as uncertain or unreliable within a text [1, 2]. They are lexical resources used by the author to indicate caution or uncertainty towards the content of the text, and in this sense they can be taken as signals of the presence of an author's opinion or attitude. Hedge cues can be expressed by several word classes: modal verbs (e.g. *can, may*), verbs (e.g. *seem, appear*), adjectives (*possibly, likely*), etc. Furthermore hedge cues can be expressed by multiword expressions, i.e. expressions that contain more than a word, with non-compositional meaning, i.e. the meaning of the expression cannot be derived from the individual meanings of the words that form the expression. This can be seen from Example 1, where *call into question* is a multiword hedge cue.

Example 1.
The low results {**call into question** the applicability of this method}.

Neither the verb *call* nor the noun *question* are hedge cues on their own, but the whole phrase conveys speculative meaning, which explains why the sentence would be marked as uncertain.

Recently, the NLP community has shown interest in problems that involve analysing language beyond the propositional meaning of sentences, i.e. the meaning in terms of truth values. Apart from performing well established NLP tasks such as parsing or semantic role labeling, there is a growing interest in tasks that involve processing non-propositional aspects of meaning, i.e. opinions, attitudes, emotions, figurative meaning. To perform these tasks, the local token-based approaches based on the lexico-syntactic features of individual words no longer suffice. The broader context of words at sentence or discourse level has to be taken into account in order to account for aspects of meaning that are expressed by certain combinations of words, like "call into question" in the sentence above. Performing hedge cue detection involves finding the linguistic expressions that express hedging. In many cases it is not possible to know whether a word belongs to a hedge cue without taking into account its context. This formed our motivation to use kLog [8], a new language for logical and relational learning with kernels. kLog is able to transform the relational representations into graph-based representations and then apply kernel methods. This makes it a suitable algorithm to process contextual aspects of language.

This paper is organized as follows. In section 2, we give an overview of related work. kLog and the modeling approach for the task at hand are presented in section 3. Section 4 discusses the experimental findings. Finally, in section 5, we conclude and present our future work.

2 Related work

Although the term *hedging* was already introduced by Lakoff in 1972 [1], and has been studied from a theoretical linguistics point since two decades [2], the interest from the computational linguistics (CL) community only arose in recent years. Light et al. [6] introduced the problem of identifying speculative language in bioscience literature. The authors use a hand-crafted list of hedge cues to identify speculative sentences in MEDLINE abstracts. They also present two systems for automatic classification of sentences in the abstracts; one based on support vector machines (SVMs), the other one based on substring matching. Medlock and Briscoe [4] extend this work and discuss the specificities of hedge classification as a weakly supervised machine learning task and present a probabilistic learning model. Furthermore they offer an improved and expanded set of annotation guidelines and provide a publicly available data set. Based on this work, Medlock [7] carried out experiments using an expanded feature space and novel representations. Szarvas [5] follows Medlock and Briscoe [4] in classifying sentences as being speculative or non-speculative. Szarvas develops a Maximum Entropy classifier that incorporates bigrams and trigrams in the feature representation and performs a reranking based feature selection procedure. Kilicoglu and Bergler [14] apply a linguistically motivated approach to the same classification task by using knowledge from existing lexical resources and incorporating syntactic patterns. Additionally, hedge cues are weighted by automatically assigning an information gain measure to them and by assigning weights semi–automatically based on their types and centrality to hedging.

Ganter and Strube [15] are the first ones in developing a system for automatic detection of sentences containing *weasels* in Wikipedia. As Ganter and Strube indicate, weasels are closely related to hedges and private states. Ganter and Strube experiment with two classifiers, one based on words preceding the weasel and another one based on syntactic patterns. The similar results of the two classifiers on sentences extracted from Wikipedia show that word frequency and distance to the weasel tag provide sufficient information. However, the classifier that uses syntactic patterns outperforms the classifier based on words on data manually re-annotated by the authors, suggesting that the syntactic patterns detect weasels that have not yet been tagged.

The increased attention for hedge detection reflects in the fact that it became a subtask of the BioNLP Shared Task in 2009 [9], and the topic of the Shared Task at CoNLL 2010 [10]. The latter comprised two levels of analysis: the focus of task 1 was learning to detect sentences containing uncertainty, whereas the objective of task 2 was resolving the in-sentence scope of hedge cues. As indicated above, the present paper will focus on task 1. As noted in [10], the approaches to this task can be classified into two major categories. Several systems approach the problem as a sentence classification problem and used a bag-of-words (BoW) feature representation. Also the individual tokens of the sentence can be classified, instead of the overall sentence. In a postprocessing step, then the sentences that contain hedge cues are classified as uncertain. In this setting, either a token classification or sequence labelling approach was taken, where in the latter the sequence information is taken into account.

3 Approach

The presented approach can be seen as a variant of the sentence classification approach that is able to represent both the lexico-syntactic information as well as the sequence information and dependency relationships in an extended feature space, which is calculated from graph kernels. This section first shortly describes kLog in section 3.1 and subsequently describes the approach taken for the hedge cue detection task (section 3.2).

3.1 kLog

kLog is a logical and relational language for kernel-based learning, that is embedded in Prolog, and builds upon and links together concepts from database theory, logic programming and learning from interpretations. It is based on a novel technique called *graphicalization* that transforms relational representations into graph based ones and derives features from a grounded entity/relationship diagram using graph kernels after which a statistical learning algorithm can be applied. The general workflow is depicted in Figure 1 and will be explained by means of the approach for the task at hand.

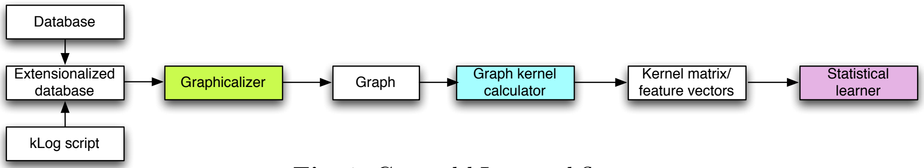

Fig. 1. General kLog workflow

3.2 Model

kLog is build upon a logical and relational data representation and is rooted in the entity-relationship (E/R) model. For the problem under consideration, the E/R-model is given in Figure 2 (left). It gives an abstract representation of the interpretations, which are sentences in the current problem. They consist of a number of consecutive words w, for which the order is represented by the $next$ relation. There are also dependency relations between certain words, that represent the structure of syntactic relations between the words of a sentence. This is modeled by $depHead$, where $depRel$ specifies the type of the dependency.

Example 2.
Often the response variable may not be continuous but rather discrete.

In Example 2 an example dependency relation exists between the determiner *the* and the noun *variable*, where the first is a noun modifier of the latter. Other properties of the word that are taken into account as features are the word string itself, its lemma, the Part-of-Speech tag (i.e. the linguistic type of the word in the sentence), the chunck tag (which indicates that a word is part of a subsequence of constituents) and a binary feature that represents whether the word is part of a predefined list of speculative strings. $weaselSentence$ represents the target relation. This E/R model representation can be transformed into a kLog script that describes (the structure of) the data. Figure 3 shows a (part of an) example interpretation z, that is a grounded version of the E/R-model, where e.g. $w(w1, `often', rb, i - dvp, 1, `often')$ specifies an entity where $w1$ is the identifier and the other attributes represent the properties. $next(w1, w2)$ gives an example relation between $w1$ and $w2$. These interpretations are then graphicalized, i.e. transformed into graphs. This can be interpreted as unfolding the E/R diagram over the data, for which an example is given in Figure 2 (left), which represents the graphicalization of the interpretation in Figure 3. This forms the input to the next level, where graph learning is applied to convert these graphicalized interpretations into extented, high-dimensional feature vectors using a graph kernel.

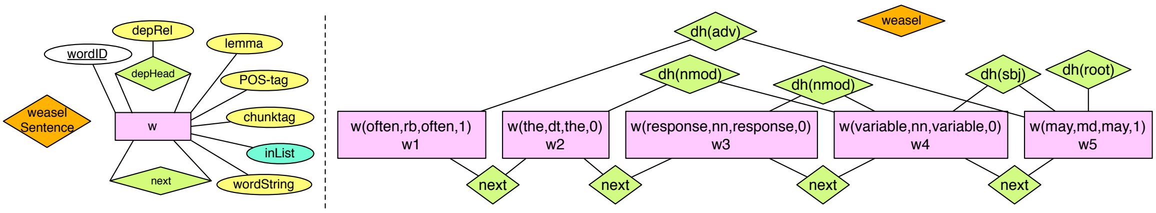

Fig. 2. Left: E/R diagram modeling the hedge cue detection task. Right: Graphicalization G_z of interpretation z (Fig. 3)

```
wwc(2).                              w(w2,'the',dt,i-np,0,'the').
next(w1,w2).                         dh(w2,w4,nmod).
w(w1,'often',rb,i-advp,1,'often').   next(w3,w4).
dh(w1,w5,adv).                       w(w3,'response',nn,i-np,0,'response').
next(w2,w3).                         dh(w3,w4,nmod).
                                     ...
```

Fig. 3. Example interpretation z

The result is a propositional learning setting, for which any statistical learner can be used. Currently, kLog employs LibSVM [11] for parameter learning.

4 Results and discussion

4.1 Dataset

For our experiments, we used the CoNLL 2010 Shared Task dataset [10] on Wikipedia, which is one of the current benchmark datasets for hedge cue resolution. The Wikipedia paragraphs were selected based on the hedge cue (called *weasels* in Wikipedia) tags that were added by the Wikipedia editors, which were subsequently manually annotated. A sentence is considered uncertain if it contains at least one weasel cue. The proportion of training and test data, and their respective class ratios can be found in Table 1.

	Train	Test
Certain	8627	7288
Uncertain	2484	2234
Total	11111	9634

Table 1. Number of instances per class in the training and test partitions of the CoNLL Shared Task Wikipedia dataset

Preprocessing For preprocessing, the approach of Morante et al. [12] was followed, in which the input files where converted into a token-per-token representation. Consequently the data was processed with the Memory Based Shallow Parser (MBSP) [13] in order to obtain lemmas, part-of-speech tags, and syntactic chunks, and with the MaltParser [16] to obtain dependency trees.

4.2 Results

The results of our approach are listed in Table 2, together with results of the 5 best listed participants in the CoNLL-Shared Task 2010. As can be noted, kLog outperforms the systems in terms of F-measure. Remarkable is that, in contrast to the other systems, kLog obtains a higher recall than precision. A possible explanation for this is that the other models applied mostly reliable patterns, whereas the relational approach in kLog is able also generalize to the less reliable ones.

Official Rank	System	P	R	F
-	**kLog**	53.9	71.6	61.5
1	Georgescul	72.0	51.7	60.2
2	Ji[1]	62.7	55.3	58.7
3	Chen	68.0	49.7	57.4
4	Morante	80.6	44.5	57.3
5	Zhang	76.6	44.4	56.2

Table 2. Evaluation performance in terms of precision, recall and F1 of the top 5 CoNLL 2010 systems and the kLog approach

[1] Remark that this system used a cross dataset approach, in which also the CoNLL 2010 biological dataset was used to train the system.

5 Conclusions and future work

In this paper we presented a new approach for solving the hedge cue resolution task, based on kernel-based logical and relational learning with kLog. Our system outperforms state-of-the-art systems, which can be ascribed to the graphicalization step, which transforms the data into a graph-based format. This enables us to use graph kernels on a full relational representation. Since the linguistic relations between words in a sentence can be represented as a graph structure, kLog seems to have the appropriate characteristics for CL problems.

In future work, we plan to test the generalizability of our approach on another dataset for this task, i.e. scientific texts from the biomedical domain, which have a different, more structured writing style and sentence structure. This opens the way to applying a cross dataset training phase, which showed improved results for one of the participants in the shared task. Furthermore kLog allows to easily redefine the problem into a sequence labeling problem, where the words are classified and the predictions for the sentences are based on the number of occurrences of words marked as hedge cues. Due to the promising result, the goal is to test this approach also on more challenging NLP problems and to perform a detailed comparison with the state-of-the-art approaches.

References

1. Lakoff, G: Hedges: A study in meaning criteria and the logic of fuzzy concepts. Journal of Philosophical Logic. 2 (1973)
2. Hyland, K.: Hedging in scientific research articles. Amsterdam (1998)
3. Riloff, E., Wiebe, J., Wilson, T.: Learning subjective nouns using extraction pattern bootstrapping. In: Proc. of CoNLL 2003. Edmonton (2003)
4. Medlock, B., Briscoe, T.: Weakly supervised learning for hedge classification in scientific literature. In: Proc. ACL 2007. Prague (2007)
5. Szarvas, G.: Hedge classification in biomedical texts with a weakly supervised selection of keywords. In: Proc. of ACL 2008. Ohio (2008).
6. Light, M., Qiu, X., Srinivasan, P.: The language of bioscience: facts, speculations, and statements in between. In: Proc. of HLT-NAACL 2004 – BioLINK. (2004).
7. Medlock, B.: Exploring hedge identification in biomedical literature. Journal of Biomedical Informatics, 41 (2008)
8. Frasconi, P., Costa F., De Raedt L., De Grave K.: kLog - a language for logical and relational learning with kernels, Technical Report, http://www.dsi.unifi.it/~paolo/ps/klog.pdf (2011)
9. Kim, J., Ohta, T., Pyysalo, S., Kano, Y., Tsujii, J.: Overview of BioNLP'09 shared task on event extraction. In: Proc. of the Workshop on Current Trends in Biomedical NLP – Shared Task. Colorado (2009)
10. Farkas, R., Vincze, V., Móra, G., Csirik, J., Szarvas, G.: The CoNLL-2010 shared task: learning to detect hedges and their scope in natural language text. In: Proc. of CoNLL 2010 – Shared Task. Uppsala (2010)
11. Chang, C.-C., Lin C.-J.: LIBSVM: a library for support vector machines (2001)
12. Morante, R., Van Asch, V., Daelemans W.: Memory-based resolution of in-sentence scopes of hedge cues. In: Proc. of CoNLL 2010 – Shared Task. Uppsala (2010)
13. Daelemans, W., van den Bosch, A.: Memory-based language processing. Cambridge University Press, Cambridge (2005)
14. Kilicoglu, H., Bergler, S.: Recognizing speculative language in biomedical research articles: a linguistically motivated perspective. In: BMC Bioinformatics (2008)
15. Ganter, V., Strube, M.: Finding hedges by chasing weasels: Hedge detection using Wikipedia tags and shallow linguistic features. In: Proc. of ACL-IJCNLP 2009 Conference Short Papers. Suntec (2009)
16. Nivre, J.: Inductive Dependency Parsing. In: Text, Speech and Language Technology. Springer (2006)