A Graph-Based Textual Entailment Method Aware of Real-World Knowledge

Saúl León, Darnes Vilariño([✉]), David Pinto, Mireya Tovar, and Beatriz Beltrán

Faculty of Computer Science, Benemérita Universidad
Autonóma de Puebla, Puebla, Mexico
{saul.leon,darnes,dpinto,mtovar,bbeltran}@cs.buap.mx
http://www.cs.buap.mx

Abstract. In this paper we propose an unsupervised methodology to solve the textual entailment task, that extracts facts associated to pair of sentences. Those extracted facts are represented as a graph. Then, two graph-based representations of two sentences may be further compared in order to determine the type of textual entailment judgment that they hold. The comparison method is based on graph-based algorithms for finding sub-graphs structures inside another graph, but generalizing the concepts by means of a real world knowledge database. The performance of the approach presented in this paper has been evaluated using the data provided in the Task 1 of the SemEval 2014 competition, obtaining 79 % accuracy.

Keywords: Textual entailment · Graph-based representation · Semantic similarity

1 Introduction

Textual entailment is defined as a process in which a directional relation between pairs of text expressions, denoted by T (the entailing "Text"), and H (the entailed "Hypothesis") is hold. We say that T entails H if the meaning of H can be inferred from the meaning of T, as would typically be interpreted by people.

Nowadays, many Natural Languaje Processing (*NLP*) tasks such as: Question Answering, Information Retrieval, Automatic Summary, Author Verification, Author Profiling, etc., usually require a module capable of detecting this type of semantic relation between a given pair of texts.

In this research work, we address the problem of textual entailment by proposing a methodology for extracting facts from a pair of sentences in order to discover the relevant information and use it for determining the textual entailment judgment between that pair of sentences. In this methodology, each set of facts associated to a sentence is represented as a graph (from the perspective of graph theory). Then, two graph-based representations of two sentences may be further compared in order to determine the type of textual entailment judgment that they hold. The comparison method is based on graph-based algorithms for

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2015, LNCS 9116, pp. 214–223, 2015.
DOI: 10.1007/978-3-319-19264-2_21

finding sub-graphs structures inside another graph, but generalizing the concepts by means of a real world knowledge database constructed on the basis of conceptnet5, wordnet and Openoffice thesaurus. The performance of the approach presented in this paper has been evaluated using the data provided in the Task 1 of the SemEval 2014 competition.

The remaining of this paper is structured as follows. In Sect. 2 we present the state of the art in textual entailment. In Sect. 3 we show our methodology based on graphs for determining the judge of textual entailment. The experimental results are presented in Sect. 4. Finally, in Sect. 5 the conclusions and further work are given.

2 Related Work

In recent years, the amount of unstructured information present in the web has overcomed our human capacity of analysis. There exist, however, a necessity of performing this process for many Natural Language Processing Tasks. With this amount of data, the analysis is a process that can only be done by using computational methods. But, the construction of automatic methods that perform this task efficiently is a big challenge that has not been solved yet.

This paper is focused in the analysis of texts for performing the particular task of textual entailment, and, therefore, we have been gathered some works reported in literature that we consider relevant for this paper.

The problem of textual entailment has been studied for more than a decade, the seminal paper know is the one published by Monz and de Rijke in 2001 [8]. This task was, thereafter, brought to a wide community in a challenge for Recognizing Textual Entailment (RTE) [2]. In general, in literature we may see that the major research works have focused their efforts in the selection of a big number of features, mainly statistical ones, that allow to build feature vectors useful in supervised learning methods for creating classification models that may further be used for determining the entailment judgment for a given new pair of sentences; some of those feature are shown in Table 1.

Presenting a comprehensive state of the art in textual entailment is not the purpose of this paper, but to present a new approach based on graph for representing information that leads to determine the entailment judgment for a pair of sentences. There exist, however, very good research works that study the trajectory of this task through the time. We refer the reader of this paper, for example, to the book published very recently by Dagan [3].

The methods reported in literature usually perform textual entailment by first extracting one or more of the aforementioned features from the two sentences that we are interested in determining the judgment. Thereafter, they use those features for constructing two feature vectors that may be compared by using basic or more complex similarity metrics, for example, those that involve semantic characteristics. This type of approaches are limited to analyze the presence or absence of common terms, but these elements are not sufficient to obtain high values of precision. A more interesting research line has been the discovering of structural patterns shared by the two sentences. For example, given the

Table 1. Enumeration of different features employed in the RTE task

Feature	Type
ngram of characters and words	Lexical
skipgram of characters and words	Lexical
Jaccard coefficient with synonyms	Lexical
Leacock &Chodorow's word similarity	Knowledge-based
Lesk's word similarity	Knowledge-based
Wu &Palmer's word similarity	Knowledge-based
Resnink's word similarity	Knowledge-based
Lin's word similarity	Knowledge-based
Jiang &Conrath's word similarity	Knowledge-based
Rada Mihalcea's metric using PMI	Corpus-based
Rada Mihalcea's metric using LSA	Corpus-based

following two sentences: *Leonardo Da Vinci painted the Mona Lisa*, and *Mona Lisa is the work of Leonardo da Vinci*, it is possible to find a structural pattern from which we can infer that *X painted Y* → *Y is the work of X*. This type of patterns may guarantee that the particular type of entailment judgment can be discovered. However, the construction/discovery of this patterns is quite difficult because the major of them are constructed for ad-hoc datasets. Real world patterns are not easy to be discovered automatically, but some efforts have been performed in this direction, in particular, seeking to generate large scale structural patterns [6].

Another research line attempt to generalize the patterns extracted for constructing axioms. This particular logic-based approach [1] seems to be promising, but still a number of problems that need to be solved, because in the major cases these approaches fail when doing the inference process. A huge amount of information is needed in order to have a proper knowledge of the real world. Even if we have properly modeled the logic representation given a set of training data, if we do not have a good knowledge database, we will usually fail to infer the real entailment judgment for the test data.

Other works in literature have tackle the RTE problem from the perspective of question-answering, thus considering that S_1 is a question and S_2 is the answer [5]. So the approaches are limited to determine certain degree of entailment based on the similarity found between the two sentences. Again, this methods may fail because even when there exist a real type of textual entailment between the sentences, some terms may not necessarily be shared by them, i.e., they are not similar from the lexical perspective, but instead they are semantically similar.

Some authors have realized that more than isoled features are needed, therefore, there have been a number of papers describing the construction of lexical resources with the aim of integrating semantic aspects in the process of textual entailment. Most of these authors use tools such as, PoS taggers, dependency parsers and other lexical resources (Wordnet, thesaurus, etc.) in the inference modules that are part of their textual entailment systems [4].

A major problem, from the machine learning perspective is the number of features of the test data that are absent in the training data. Normally, this problem may be overcomed by introducing more data in the training set, or using some kind or smoothing techniques. But, most of the time, the fact that some particular term is not found in the test data is a consequence of a lack of real world knowledge, for example, the term is a hyperonym, hyponym, synonym, etc. Integrating all these features in basic text representation schemas is not so simple, especially if we are interested in conserving the knowledge present in the original sentence. In this way, graph structures are a natural way to represent natural language information, preserving different level of natural language formal description (lexical, syntax, semantics, etc.). There are papers such as: the work [9] that presents statistical machine learning representation of textual entailment via syntactic graphs constituted by tree pairs. They shows that the natural way of representing the syntactic relation between text and hypothesis consists in the huge feature space of all possible syntactic tree fragment pairs, which can only be managed using kernel methods. Moreover the author Okita [10] presents a robust textual entailment system using the principle of just noticeable difference in psychology, which they call a local graph matching-based system with active learning.

Thus, we consider important to use both, graph based structures for preserving the original structure of the sentence and, knowledge databases that may allow our methods to be aware of those lexical relation hold in the terms with the purpose of having more efficient methods for the automatic recognition of textual entailment.

3 A Methodology Based on Graphs Aware of Real World Relation for Textual Entailment

As we mentioned before, we are proposing a method that requires to be aware that some terms are semantically related, a process that require to construct a process for calculating this type of similarity which in fact is much more difficult than calculating lexical similarity. We need, therefore, to use a proper lexical resource for improving the calculation of the degree of semantic similarity between sentences. For this goal, we have constructed a general purpose knowledge database on the basis of the following lexical resources: WordNet[1], ConceptNet5[2] and OpenOffice Thesaurus[3]. The manner we have used this database to infer real-world knowledge with the aim of using semantic information in the textual entailment task, we describe this in the following subsection.

[1] http://wordnet.princeton.edu/.

[2] http://conceptnet5.media.mit.edu/.

[3] https://wiki.openoffice.org/wiki/Dictionaries.

3.1 Inference Through Knowledge Databases Using Graph-Based Representation

Sometimes, two terms are semantically related by a given type of relation, let us say, synonym, hyperonym, etc. For example, from the real-world knowledge we know that a "camera" is an "electronic device". In this case, we would like to be able to automatically determine this type of relation. There exist, however other cases in which the relation is not hold directly, but by using graph searching processes we can infer these relation. For example, the terms "head" and "hair" are semantically related, but this relation can not be found directly from our knowledge database. However, we can find and indirect relation between these two words, as shown in Fig. 1, because from the same database we know that "head" is related with "body structure", "body structure" is related with "filament", and "filament" is related with "hair", therefore, both "head" and "hair" are indirectly related by transitivity.

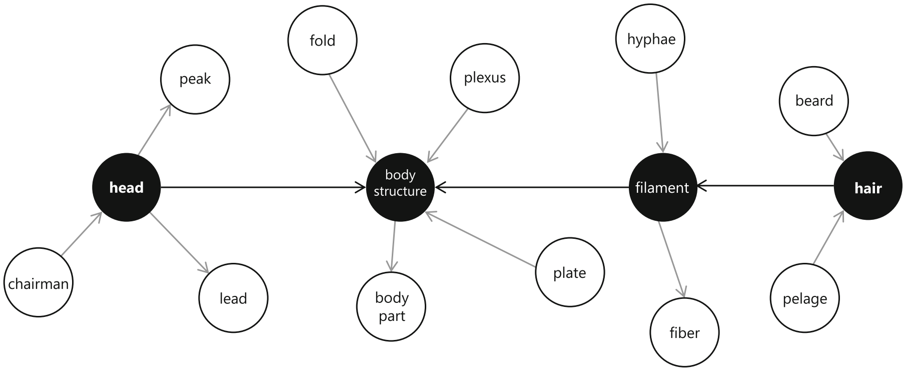

Fig. 1. Concepts network that relates head with hair

This process may be done because we have a knowledge database in which we may execute single queries with the aim of determining whether or not a given pair of concepts are related. Formally, the knowledge database can be seen as a concept graph $G_C = (V, E)$, where V is the set of vertices (concepts), and E is the set of edges (relations). There may be a lot of relations, but for practical purposes we have employed only the following ones: PartOf, Synonym, RelatedTo, WordNetRelatedTo, MemberOf, SimilarTo.

We consider that by adding this process to the methodology we can better accomplish the final purpose of this research work that is to automatically detect the judge of textual entailment for a given pair of sentences.

Having a mechanism for detecting semantic relations, we now propose to employ graph structures for representing sentences through "facts" with the aim of preserving different levels of formal description of language in a single structure. A fact is basically a relation between two language components (Noun phrase, Verb or Adjective); the relation may be one of the following five types: subject, object, qualify, extension or complement. A better description of the facts constructing process follows.

3.2 Building Facts by Sentence Interpretation

In order to build facts, we need to transform the original raw sentence to a one with tags that allow us to interpret the role that each language component plays in the sentence. We propose to start by using the Stanford parser[4]. By using this tool we may obtain PoS tags, the parse and the typed dependencies. The parsing process identify the following chunk tags: Noun Phrases (NP), Verbal Phrases (VP), and Prepositional Phrase (PP), which allow us to generate the following four types of categories[5]:

- *ENTITY*: Nouns in a noun phrase associated to one of the following PoS tags: NN, NNS, NNP, NNPS and NAC.
- *ACTIVITY*: Verbs in a verbal phrase associated to one of the following PoS tags: VB, VBG, VBZ, VBP, VBN, VBD and MD.
- *QUALITY*: Adjectives in a noun phrase associated to one of the following PoS tags: JJ, JJR and JJS.
- *PREPOSITION*: Prepositions in a prepositional phrase associated to one of the following PoS tags: IN and TO.

From the previously detected categories we may discover the following facts:

- *Subject*: This fact is obtained when a given ENTITY is associated to one or more ACTIVITY. We assume that such ENTITY is the “subject” of those activities.
- *Object*: This fact is hold when a given ACTIVITY is associated to one or more ENTITY. In this case, the ENTITY is the “object” of that ACTIVITY.
- *Qualify*: This fact is detected when a given QUALIFY is associated to one ENTITY, thus the QUALITY “qualify” the ENTITY.
- *Extension*: This fact is obtained when a given ENTITY is associated to one PREPOSITION. In this case, we say that the ENTITY has an “extension”.
- *Complement*: This fact is given when a given PREPOSITION is associated to one ENTITY. The fact indicate that this ENTITY is a “complement” of a another ENTITY.

The automatic generation of the facts previously mentioned lead us to the rules shown in Table 2, that allow us to extract these facts from the parsed version of the sentences.

We observe this four step process which traduces the original sentence to a set of facts which may be further used in the process of recognizing textual entailment.

[4] http://nlp.stanford.edu/software/lex-parser.shtml.

[5] The names have been proposed with the aim of giving some sense to each type of category.

Table 2. Rules for the automatic extraction of facts

Rule	Fact
$X \rightarrow ENTITY$ && $X \rightarrow ACTIVITY$	subject($ENTITY$, $ACTIVITY$)
$ACTIVITY \rightarrow ENTITY$	object($ACTIVITY$, $ENTITY$)
$X \rightarrow ENTITY$ && $X \rightarrow QUALIFY$	qualify($QUALIFY$, $ENTITY$)
$ENTITY \rightarrow PREPOSITION$	extension($ENTITY$, $PREPOSITION$)
$PREPOSITION \rightarrow ENTITY$	complement($PREPOSITION$, $ENTITY$)

3.3 Recognition of Textual Entailment

We have extracted facts with the aim of giving an interpretation to the sense of each sentence. However, isolated facts are not as useful as when they are brought together, therefore, we use graph structures for preserving the richness of all these facts. Graphs are a non-linear structure that allow us to relate, in this case, concepts for understanding the manner this relation acts over those concepts. These structures makes it possible to calculate the semantic similarity between pair of sentences, and eventually to describe rules for interpreting those similarities as textual entailment judgment criteria.

Formally, a set of facts is represented as a labeled graph $G = (V, E, L_E, \beta)$, where:

- $V = \{v_i | i = 1, ..., n\}$ is a finite set of vertices, $V \neq \emptyset$, and $n = \#vertices$ in the graph.
- $E \subseteq V \times V$ is the finite set of edges, $E = \{e = \{v_i, v_j\} | v_i, v_j \in V, 1 \leq i, j \leq n\}$.
- L_E, is the fact relation.
- $\beta : E \rightarrow L_E$, is a function that assigns a fact relationship to the edge.

Once the graphs are constructed we are able to use graph-based algorithms for finding any type of semantic relation among different sentences. For example, we could find the lexical similarity, or even semantic similarity by using the real-world knowledge database previously constructed, thus enriching the process of detecting the judgement of textual entailment.

We propose an approach in which the textual entailment task is solved analyzing the number of facts shared by the two graphs that represent both the entailing "Text" T, and the entailed "Hypothesis" H. We perform this process by eliminating from both graphs the substructures shared; in this way, the number of graph structures both graphs hold after removing similar substructures allow us to detect the occurrence of textual entailment.

In order to analyze the obtained graph substructures, it is not sufficient to directly compare ENTITIES and ACTIVITIES, because it is needed to detect how two ENTITIES interact given an ACTIVITY, therefore, instead of using a single fact, we use a substructure made up of three concepts linked each pair of concepts by a given relation. We named this type of substructure as an EXPANDED fact (EXP_FACT). The construction of the EXP_FACT are shown in Table 3.

Table 3. The three rules for generating expanded facts

Facts	EXP_FACT
(1) $subject(ENTITY_1, ACTIVITY)$ AND $object(ACTIVITY, ENTITY_2)$	$\rightarrow (ENTITY_1,\ ACTIVITY,\ ENTITY_2)$
(2) $qualify(QUALITY, ENTITY)$	$\rightarrow (ENTITY,$ qualify, $QUALITY)$
(3) $extension(ENTITY_1, PREPOSITION)$ AND $complement(PREPOSITION, ENTITY_2)$	$\rightarrow (ENTITY_1,\ PREPOSITION,\ ENTITY_2)$

This manner of analyzing the relation among facts allow us to detect the main ideas that are transmitted in each sentence.

The number of original concepts in the graphs is compared after removing similar EXP_FACT substructures in both graphs (G_1 and G_2). If the proportion of concepts removed is greater than 50 % in the two graphs, then we can say that there exist textual entailment between the two sentences. Otherwise, we can not determine the entailment judgment, so we establish it as "neutral".

4 Experimental Results

In order to have a comparison of the performance of the proposed methodology, we have used the dataset given at the subtask 2 of Task 1 of the SemEval 2014 conference[6]. In that subtask, it is required to solve the problem of textual entailment between two sentences S_1 and S_2 by automatically detecting one of the following judgments:

- *ENTAILMENT*: if S_1 entails S_2
- *CONTRADICTION*: if S_1 contradicts S_2
- *NEUTRAL*: the truth of S_1 cannot be determined on the basis of S_2

The methodology proposed is not able to detect the CONTRADICTION judgment, but only ENTAILMENT or NEUTRAL judgment. Therefore, we propose to use a simple technique based on the existence of cuewords associated to antonyms or negation. Thus, in the pair of sentences S_1 and S_2, if one of this contains negation words or antonyms means that a CONTRADICTION exists.

In the following section we describe the dataset used in the experiments.

4.1 Test Dataset

The test dataset used at SemEVal 2014 was constructed as mentioned in [7]. The description of the number of samples for each type of textual entailment judgment is given in Table 4. Since the dataset is manually annotated, we are able to calculate the performance of the methodology we propose in this paper.

[6] International Workshop on Semantic Evaluation.

Table 4. Judgments distributions

Judgment	Test datasetd
NEUTRAL	2793
ENTAILMENT	1414
CONTRADICTION	720

4.2 Obtained Results

It is worth mentioned that the methodology proposed is an unsupervised approach, since the perspective that we do not a training corpus. We only use an external resource (knowledge database) for determining semantic similarities among concepts. In Table 5 we can see the results obtained by other research teams at the SemEval 2014 competition. There we have named **BUAP GraphMatch** to the approach presented in this paper. As we can see, the performance is similar to other approaches submitted to the competition with and accuracy of 79 %. The main difference here is , as we already mentioned, that we do not need to construct a classification model because our approach is unsupervised. We can move to different domains and the performance should not be affected as the target domain can be modeled in some way by the relation stored in the knowledge database.

Table 5. Task1, subtask 2, Semeval 2014 results

Team ID	Accuracy
Illinois-LH_run1	84.575
ECNU_run1	83.641
UNAL-NLP_run1	83.053
SemantiKLUE_run1	82.322
The_Meaning_Factory_run1	81.591
CECL_ALL_run1	79.988
BUAP_run1	79.663
BUAP GraphMatch	**79.072**

5 Conclusions

The methodology proposed for textual entailment aims to use graph structures for representing expanded facts, which contain information of the real world. This unsupervised approach performed well on the textual entailment task with a performance of 79 %. We consider that we have exploited the graph representation by (1) infering transitive relation between pair of concepts, and (2) detecting graph substructures with the aim of determining a textual entailment judgment.

We have noticed that the process of inference helps to improve the performance of the methodology, but a deep analysis over this process need to be done in the future, because we have observed that the semantic similarity between concepts may be degraded when the number of intermediate concepts is too high.

We consider that given the level of complexity of the task carried out in this paper, the methodology is attractive. It should be very interesting to test the same methodology in other natural language task such as summarization, information retrieval, question-answering, etc.

As future work we plan to use other types of graph-based algorithms for calculating similarities. We would also investigate the result of enriching the knowledge database used in the inference process.

References

1. Bos, J.: Is there a place for logic in recognizing textual entailment? Linguist. Issues Lang. Technol. **9**(3), 1–18 (2013)
2. Dagan, I., Glickman, O., Magnini, B.: The pascal recognising textual entailment challenge. In: Quiñonero-Candela, J., Dagan, I., Magnini, B., d'Alché-Buc, F. (eds.) Machine learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment. LNCS, vol. 3944, pp. 177–190. Springer, Heidelberg (2006)
3. Dagan, I., Roth, D., Sammons, M., Zanzotto, F.M.: Recognizing textual entailment: models and applications. Synth. Lect. Hum. Lang. Technol. **6**(4), 1–220 (2013)
4. Day, M.-Y., Tu, C.,Huang, S.-J., Vong, H.C., Wu, S.-W.: IMTKU textual entailment system for recognizing inference in text at NTCIR-10 RITE-2. In: Proceedings of the 10th NTCIR Conference (2013)
5. Harabagiu, S., Hickl, A.: Methods for using textual entailment in open-domain question answering. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics, pp. 905–912. Association for Computational Linguistics (2006)
6. Kouylekov, M., Magnini, B.: Building a large-scale repository of textual entailment rules. In: Proceedings of LREC (2006)
7. Marelli, M., Menini, S., Baroni, M., Bentivogli, L., Bernardi, R., Zamparelli, R.: A sick cure for the evaluation of compositional distributional semantic models. In: Proceedings of LREC (2014)
8. Monz, C., de Rijke, M.: Tequesta: the university of amsterdam's textual question answering system. In: TREC (2001)
9. Moschitti, A., Zanzotto, F.: Encoding tree pair -based graphs in learning algorithms: the textual entailment, pp. 25–32 (2008)
10. Tsuyoshi, O.: Active learning-based local graph matching for textual entailment (2013)