Investigating a Generic Paraphrase-based Approach for Relation Extraction

Lorenza Romano
ITC-irst
via Sommarive, 18
38050 Povo (TN), Italy
romano@itc.it

Milen Kouylekov
ITC-irst
via Sommarive, 18
38050 Povo (TN), Italy
kouylekov@itc.it

Idan Szpektor
Department of Computer Science
Bar Ilan University
Ramat Gan, 52900, Israel
szpekti@cs.biu.ac.il

Ido Dagan
Department of Computer Science
Bar Ilan University
Ramat Gan, 52900, Israel
dagan@cs.biu.ac.il

Alberto Lavelli
ITC-irst
via Sommarive, 18
38050 Povo (TN), Italy
lavelli@itc.it

Abstract

Unsupervised paraphrase acquisition has been an active research field in recent years, but its effective coverage and performance have rarely been evaluated. We propose a generic paraphrase-based approach for Relation Extraction (RE), aiming at a dual goal: obtaining an applicative evaluation scheme for paraphrase acquisition and obtaining a generic and largely unsupervised configuration for RE. We analyze the potential of our approach and evaluate an implemented prototype of it using an RE dataset. Our findings reveal a high potential for unsupervised paraphrase acquisition. We also identify the need for novel robust models for matching paraphrases in texts, which should address syntactic complexity and variability.

1 Introduction

A crucial challenge for semantic NLP applications is recognizing the many different ways for expressing the same information. This *semantic variability* phenomenon was addressed within specific applications, such as question answering, information extraction and information retrieval. Recently, the problem was investigated within generic application-independent paradigms, such as paraphrasing and textual entailment.

Eventually, it would be most appealing to apply generic models for semantic variability to concrete applications. This paper investigates the applicability of a generic "paraphrase-based" approach to the Relation Extraction (RE) task, using an available RE dataset of protein interactions. RE is highly suitable for such investigation since its goal is to exactly identify all the different variations in which a target semantic relation can be expressed. Taking this route sets up a dual goal: (a) from the generic paraphrasing perspective - an objective evaluation of paraphrase acquisition performance on a concrete application dataset, as well as identifying the additional mechanisms needed to match paraphrases in texts; (b) from the RE perspective - investigating the feasibility and performance of a generic paraphrase-based approach for RE.

Our configuration assumes a set of entailing templates (non-symmetric "paraphrases") for the target relation. For example, for the target relation "X interact with Y" we would assume a set of entailing templates as in Tables 3 and 7. In addition, we require a syntactic matching module that identifies template instances in text.

First, we manually analyzed the protein-interaction dataset and identified all cases in which protein interaction is expressed by an entailing template. This set a very high idealized upper bound for the recall of the paraphrase-based approach for this dataset. Yet, obtaining high coverage in practice would require effective paraphrase acquisition and lexical-syntactic template matching. Next, we implemented a prototype that utilizes a state-of-the-art method for learning entailment relations from the web (Szpektor et al., 2004), the Minipar dependency parser (Lin, 1998) and a syntactic matching module. As expected, the performance of the implemented system was much lower than the ideal upper bound, yet obtaining quite reasonable practical results given its unsupervised nature.

The contributions of our investigation follow the dual goal set above. To the best of our knowledge, this is the first comprehensive evaluation that measures directly the performance of unsupervised paraphrase acquisition relative to a standard application dataset. It is also the first evaluation of a generic paraphrase-based approach for the standard RE setting. Our findings are encouraging for both goals, particularly relative to their early maturity level, and reveal constructive evidence for the remaining room for improvement.

2 Background

2.1 Unsupervised Information Extraction

Information Extraction (IE) and its subfield Relation Extraction (RE) are traditionally performed in a supervised manner, identifying the different ways to express a specific information or relation. Given that annotated data is expensive to produce, unsupervised or weakly supervised methods have been proposed for IE and RE.

Yangarber et al. (2000) and Stevenson and Greenwood (2005) define methods for automatic acquisition of predicate-argument structures that are similar to a set of seed relations, which represent a specific scenario. Yangarber et al. (2000) approach was evaluated in two ways: (1) manually mapping the discovered patterns into an IE system and running a full MUC-style evaluation; (2) using the learned patterns to perform document filtering at the scenario level. Stevenson and Greenwood (2005) evaluated their method through document and sentence filtering at the scenario level.

Sudo et al. (2003) extract dependency subtrees within relevant documents as IE patterns. The goal of the algorithm is event extraction, though performance is measured by counting argument entities rather than counting events directly.

Hasegawa et al. (2004) performs unsupervised hierarchical clustering over a simple set of features. The algorithm does not extract entity pairs for a given relation from a set of documents but rather classifies all relations in a large corpus. This approach is more similar to text mining tasks than to classic IE problems.

To conclude, several unsupervised approaches learn relevant IE templates for a complete scenario, but without identifying their relevance to each specific relation within the scenario. Accordingly, the evaluations of these works either did not address the direct applicability for RE or evaluated it only after further manual postprocessing.

2.2 Paraphrases and Entailment Rules

A generic model for language variability is using paraphrases, text expressions that roughly convey the same meaning. Various methods for automatic paraphrase acquisition have been suggested recently, ranging from finding equivalent lexical elements to learning rather complex paraphrases at the sentence level[1].

More relevant for RE are "atomic" paraphrases between *templates*, text fragments containing variables, e.g. 'X buy $Y \Leftrightarrow X$ purchase Y'. Under a syntactic representation, a template is a parsed text fragment, e.g. '$X \overset{subj}{\leftarrow}$ interact $\overset{mod}{\rightarrow}$ with $\overset{pcomp-n}{\rightarrow} Y$' (based on the syntactic dependency relations of the Minipar parser). The parses include part-of-speech tags, which we omit for clarity.

Dagan and Glickman (2004) suggested that a somewhat more general notion than paraphrasing is that of *entailment relations*. These are directional relations between two templates, where the meaning of one can be entailed from the meaning of the other, e.g. 'X bind to $Y \Rightarrow X$ interact with Y'. For RE, when searching for a target relation, it is sufficient to identify an entailing template since it implies that the target relation holds as well. Under this notion, paraphrases are bidirectional entailment relations.

Several methods extract atomic paraphrases by exhaustively processing local corpora (Lin and Pantel, 2001; Shinyama et al., 2002). Learning from a local corpus is bounded by the corpus scope, which is usually domain specific (both works above processed news domain corpora). To cover a broader range of domains several works utilized the Web, while requiring several manually provided examples for each input relation, e.g. (Ravichandran and Hovy, 2002). Taking a step further, the TEASE algorithm (Szpektor et al., 2004) provides a completely unsupervised method for acquiring entailment relations from the Web for a given input relation (see Section 5.1).

Most of these works did not evaluate their results in terms of application coverage. Lin and Pantel (2001) compared their results to human-generated paraphrases. Shinyama et al. (2002) measured the coverage of their learning algorithm relative to the paraphrases present in a given corpus. Szpektor et al. (2004) measured "yield", the number of correct rules learned for an input re-

[1] See the 3rd IWP workshop for a sample of recent works on paraphrasing (http://nlp.nagaokaut.ac.jp/IWP2005/).

lation. Ravichandran and Hovy (2002) evaluated the performance of a QA system that is based solely on paraphrases, an approach resembling ours. However, they measured performance using Mean Reciprocal Rank, which does not reveal the actual coverage of the learned paraphrases.

3 Assumed Configuration for RE

Phenomenon	Example
Passive form	'*Y* is activated by *X*'
Apposition	'*X* activates its companion, *Y*'
Conjunction	'*X* activates prot3 and *Y*'
Set	'*X* activates two proteins, *Y* and *Z*'
Relative clause	'*X*, which activates *Y*'
Coordination	'*X* binds and activates *Y*'
Transparent head	'*X* activates a fragment of *Y*'
Co-reference	'*X* is a kinase, though it activates *Y*'

Table 1: Syntactic variability phenomena, demonstrated for the normalized template '*X* activate *Y*'.

The general configuration assumed in this paper for RE is based on two main elements: a list of lexical-syntactic templates which entail the relation of interest and a syntactic matcher which identifies the template occurrences in sentences. The set of entailing templates may be collected either manually or automatically. We propose this configuration both as an algorithm for RE and as an evaluation scheme for paraphrase acquisition.

The role of the syntactic matcher is to identify the different syntactic variations in which templates occur in sentences. Table 1 presents a list of generic syntactic phenomena that are known in the literature to relate to linguistic variability. A phenomenon which deserves a few words of explanation is the "transparent head noun" (Grishman et al., 1986; Fillmore et al., 2002). A transparent noun *N1* typically occurs in constructs of the form '*N1 preposition N2*' for which the syntactic relation involving *N1*, which is the head of the NP, applies to *N2*, the modifier. In the example in Table 1, 'fragment' is the transparent head noun while the relation 'activate' applies to *Y* as object.

4 Manual Data Analysis

4.1 Protein Interaction Dataset

Bunescu et al. (2005) proposed a set of tasks regarding protein name and protein interaction extraction, for which they manually tagged about 200 Medline abstracts previously known to contain human protein interactions (a binary symmetric relation). Here we consider their RE task of extracting interacting protein pairs, given that the correct protein names have already been identified. All protein names are annotated in the given gold standard dataset, which includes 1147 annotated interacting protein pairs. Protein names are rather complex, and according to the annotation adopted by Bunescu et al. (2005) can be substrings of other protein names (e.g., `<prot> <prot> GITR </prot> ligand </prot>`). In such cases, we considered only the longest names and protein pairs involving them. We also ignored all reflexive pairs, in which one protein is marked as interacting with itself. Altogether, 1052 interactions remained. All protein names were transformed into symbols of the type Prot$N$, where N is a number, which facilitates parsing.

For development purposes, we randomly split the abstracts into a 60% development set (575 interactions) and a 40% test set (477 interactions).

4.2 Dataset analysis

In order to analyze the potential of our approach, two of the authors manually annotated the 575 interacting protein pairs in the development set. For each pair the annotators annotated whether it can be identified using only template-based matching, assuming an ideal implementation of the configuration of Section 3. If it can, the *normalized form* of the template connecting the two proteins was annotated as well. The normalized template form is based on the active form of the verb, stripped of the syntactic phenomena listed in Table 1. Additionally, the relevant syntactic phenomena from Table 1 were annotated for each template instance. Table 2 provides several example annotations.

A Kappa value of 0.85 (*nearly perfect agreement*) was measured for the agreement between the two annotators, regarding whether a protein pair can be identified using the template-based method. Additionally, the annotators agreed on 96% of the normalized templates that should be used for the matching. Finally, the annotators agreed on at least 96% of the cases for each syntactic phenomenon except transparent heads, for which they agreed on 91% of the cases. This high level of agreement indicates both that template-based matching is a well defined task and that normalized template form and its syntactic variations are well defined notions.

Several interesting statistics arise from the an-

Sentence	Annotation
We have crystallized a complex between human **FGF1** and a two-domain extracellular fragment of human **FGFR2**.	• template: ‘complex between *X* and *Y*’ • transparent head: ‘fragment of *X*’
CD30 and its counter-receptor CD30 ligand (**CD30L**) are members of the TNF-receptor / TNFalpha superfamily and function to regulate lymphocyte survival and differentiation.	• template: ‘*X*’s counter-receptor *Y*’ • apposition • co-reference
iCdi1, a human G1 and S phase protein phosphatase that associates with **Cdk2**.	• template: ‘*X* associate with *Y*’ • relative clause

Table 2: Examples of annotations of interacting protein pairs. The annotation describes the normalized template and the different syntactic phenomena identified.

Template	*f*	Template	*f*	Template	*f*
X interact with *Y*	28	interaction of *X* with *Y*	12	*X Y* interaction	5
X bind to *Y*	22	*X* associate with *Y*	11	*X* interaction with *Y*	4
X Y complex	17	*X* activate *Y*	6	association of *X* with *Y*	4
interaction between *X* and *Y*	16	binding of *X* to *Y*	5	*X*’s association with *Y*	3
X bind *Y*	14	*X* form complex with *Y*	5	*X* be agonist for *Y*	3

Table 3: The 15 most frequent templates and their instance count (*f*) in the development set.

notation. First, 93% of the interacting protein pairs (537/575) can be potentially identified using the template-based approach, if the relevant templates are provided. This is a very promising finding, suggesting that the template-based approach may provide most of the requested information. We term these 537 pairs as *template-based pairs*. The remaining pairs are usually expressed by complex inference or at a discourse level.

Phenomenon	%	Phenomenon	%
transparent head	34	relative clause	8
apposition	24	co-reference	7
conjunction	24	coordination	7
set	13	passive form	2

Table 4: Occurrence percentage of each syntactic phenomenon within template-based pairs (537).

Second, for 66% of the template-based pairs at least one syntactic phenomenon was annotated. Table 4 contains the occurrence percentage of each phenomenon in the development set. These results show the need for a powerful syntactic matcher on top of high performance template acquisition, in order to correctly match a template in a sentence.

Third, 175 different normalized templates were identified. For each template we counted its *template instances*, the number of times the template occurred, counting only occurrences that express an interaction of a protein pair. In total, we counted 341 template instances for all 175 templates. Interestingly, 50% of the template instances (184/341) are instances of the 21 most frequent templates. This shows that, though protein interaction can be expressed in many ways, writers tend to choose from among just a few common expressions. Table 3 presents the most frequent templates. Table 5 presents the minimal number of templates required to obtain the range of different recall levels.

Furthermore, we grouped template variants that are based on morphological derivations (e.g. ‘*X* interact with *Y*’ and ‘*X Y* interaction’) and found that 4 groups, ‘*X* interact with *Y*’, ‘*X* bind to *Y*’, ‘*X* associate with *Y*’ and ‘*X* complex with *Y*’, together with their morphological derivations, cover 45% of the template instances. This shows the need to handle generic lexical-syntactic phenomena, and particularly morphological based variations, separately from the acquisition of normalized lexical syntactic templates.

To conclude, this analysis indicates that the template-based approach provides very high coverage for this RE dataset, and a small number of normalized templates already provides significant recall. However, it is important to (a) develop a model for morphological-based template variations (e.g. as encoded in Nomlex (Macleod et al.,)), and (b) apply accurate parsing and develop syntactic matching models to recognize the rather complex variations of template instantiations in text. Finally, we note that our particular figures are specific to this dataset and the biological abstracts domain. However, the annotation and analysis methodologies are general and are suggested as highly effective tools for further research.

R(%)	# templates	R(%)	# templates
10	2	60	39
20	4	70	73
30	6	80	107
40	11	90	141
50	21	100	175

Table 5: The number of most frequent templates necessary to reach different recall levels within the 341 template instances.

5 Implemented Prototype

This section describes our initial implementation of the approach in Section 3.

5.1 TEASE

The TEASE algorithm (Szpektor et al., 2004) is an unsupervised method for acquiring entailment relations from the Web for a given input template. In this paper we use TEASE for entailment relation acquisition since it processes an input template in a completely unsupervised manner and due to its broad domain coverage obtained from the Web. The reported percentage of correct output templates for TEASE is 44%.

The TEASE algorithm consists of 3 steps, demonstrated in Table 6. TEASE first retrieves from the Web sentences containing the input template. From these sentences it extracts variable instantiations, termed *anchor-sets*, which are identified as being characteristic for the input template based on statistical criteria (first column in Table 6). Characteristic anchor-sets are assumed to uniquely identify a specific event or fact. Thus, any template that appears with such an anchor-set is assumed to have an entailment relationship with the input template. Next, TEASE retrieves from the Web a corpus S of sentences that contain the characteristic anchor-sets (second column), hoping to find occurrences of these anchor-sets within templates other than the original input template. Finally, TEASE parses S and extracts templates that are assumed to entail or be entailed by the input template. Such templates are identified as *maximal most general* sub-graphs that contain the anchor sets' positions (third column in Table 6). Each learned template is ranked by number of occurrences in S.

5.2 Transformation-based Graph Matcher

In order to identify instances of entailing templates in sentences we developed a syntactic matcher that is based on transformations rules. The matcher processes a sentence in 3 steps: 1) parsing the sentence with the Minipar parser, obtaining a dependency graph[2]; 2) matching each template against the sentence dependency graph; 3) extracting candidate term pairs that match the template variables.

A template is considered directly matched in a sentence if it appears as a sub-graph in the sentence dependency graph, with its variables instantiated. To further address the syntactic phenomena listed in Table 1 we created a set of hand-crafted parser-dependent *transformation rules*, which account for the different ways in which syntactic relationships may be realized in a sentence. A transformation rule maps the left hand side of the rule, which strictly matches a sub-graph of the given template, to the right hand side of the rule, which strictly matches a sub-graph of the sentence graph. If a rule matches, the template sub-graph is mapped accordingly into the sentence graph.

For example, to match the syntactic template '*X*(N) $\overset{subj}{\leftarrow}$ activate(V) $\overset{obj}{\rightarrow}$ *Y*(N)' (POS tags are in parentheses) in the sentence "*Prot1 detected and activated Prot2*" (see Figure 1) we should handle the coordination phenomenon. The matcher uses the transformation rule '$\underline{\textit{Var1}}$(V) $\Rightarrow$ and(U)$\overset{mod}{\leftarrow}$ $\underline{\textit{Word}}$(V) $\overset{conj}{\rightarrow}$ *Var1*(V)' to overcome the syntactic differences. In this example *Var1* matches the verb 'activate', *Word* matches the verb 'detect' and the syntactic relations for *Word* are mapped to the ones for *Var1*. Thus, we can infer that the subject and object relations of 'detect' are also related to 'activate'.

6 Experiments

6.1 Experimental Settings

To acquire a set of entailing templates we first executed TEASE on the input template '*X* $\overset{subj}{\leftarrow}$ interact $\overset{mod}{\rightarrow}$ with $\overset{pcomp-n}{\rightarrow}$ *Y*', which corresponds to the "default" expression of the protein interaction

[2] We chose a dependency parser as it captures directly the relations between words; we use Minipar due to its speed.

Extracted Anchor-set	Sentence containing Anchor-set	Learned Template
X=‘chemokines’, Y=‘specific receptors’	Chemokines bind to specific receptors on the target cells	$X \overset{subj}{\longleftarrow}$ bind $\overset{mod}{\longrightarrow}$ to $\overset{pcomp-n}{\longrightarrow} Y$
X=‘Smad3’, Y=‘Smad4’	Smad3 / Smad4 complexes translocate to the nucleus	$X\ Y \overset{nn}{\longrightarrow}$ complex

Table 6: TEASE output at different steps of the algorithm for ‘$X \overset{subj}{\longleftarrow}$ interact $\overset{mod}{\longrightarrow}$ with $\overset{pcomp-n}{\longrightarrow} Y$’.

1. *X bind to Y*	7. *X Y complex*	13. *X interaction with Y*
2. *X activate Y*	8. *X recognize Y*	14. *X trap Y*
3. *X stimulate Y*	9. *X block Y*	15. *X recruit Y*
4. *X couple to Y*	10. *X binding to Y*	16. *X associate with Y*
5. *interaction between X and Y*	11. *X Y interaction*	17. *X be linked to Y*
6. *X become trapped in Y*	12. *X attach to Y*	18. *X target Y*

Table 7: The top 18 correct templates learned by TEASE for ‘*X* interact with *Y*’.

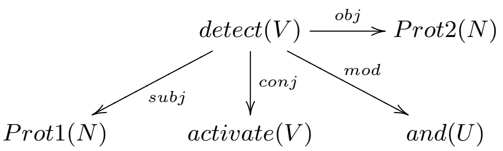

Figure 1: The dependency parse graph of the sentence “*Prot1 detected and activated Prot2*”.

relation. TEASE learned 118 templates for this relation. Table 7 lists the top 18 learned templates that we considered as correct (out of the top 30 templates in TEASE output). We then extracted interacting protein pair candidates by applying the syntactic matcher to the 119 templates (the 118 learned plus the input template). Candidate pairs that do not consist of two proteins, as tagged in the input dataset, were filtered out (see Section 4.1; recall that our experiments were applied to the dataset of protein interactions, which isolates the RE task from the protein name recognition task).

In a subsequent experiment we iteratively executed TEASE on the 5 top-ranked learned templates to acquire additional relevant templates. In total, we obtained 1233 templates that were likely to imply the original input relation. The syntactic matcher was then reapplied to extract candidate interacting protein pairs using all 1233 templates.

We used the development set to tune a small set of 10 generic hand-crafted transformation rules that handle different syntactic variations. To handle transparent head nouns, which is the only phenomenon that demonstrates domain dependence, we extracted a set of the 5 most frequent transparent head patterns in the development set, e.g. ‘fragment of *X*’.

In order to compare (roughly) our performance with supervised methods applied to this dataset, as summarized in (Bunescu et al., 2005), we adopted their recall and precision measurement. Their scheme counts over distinct protein pairs per abstract, which yields 283 interacting pairs in our test set and 418 in the development set.

6.2 Manual Analysis of TEASE Recall

experiment	pairs	instances
input	39%	37%
input + iterative	49%	48%
input + iterative + morph	63%	62%

Table 8: The potential recall of TEASE in terms of distinct pairs (out of 418) and coverage of template instances (out of 341) in the development set.

Before evaluating the system as a whole we wanted to manually assess in isolation the coverage of TEASE output relative to all template instances that were manually annotated in the development set. We considered a template as covered if there is a TEASE output template that is equal to the manually annotated template or differs from it only by the syntactic phenomena described in Section 3 or due to some parsing errors. Counting these matches, we calculated the number of template instances and distinct interacting protein pairs that are covered by TEASE output.

Table 8 presents the results of our analysis. The

1st line shows the coverage of the 119 templates learned by TEASE for the input template '*X* interact with *Y*'. It is interesting to note that, though we aim to learn relevant templates for the specific domain, TEASE learned relevant templates also by finding anchor-sets of different domains that use the same jargon, such as particle physics.

We next analyzed the contribution of the iterative learning for the additional 5 templates to recall (2nd line in Table 8). With the additional learned templates, recall increased by about 25%, showing the importance of using the iterative steps.

Finally, when allowing matching between a TEASE template and a manually annotated template, even if one is based on a morphological derivation of the other (3rd line in Table 8), TEASE recall increased further by about 30%.

We conclude that the potential recall of the current version of TEASE on the protein interaction dataset is about 60%. This indicates that significant coverage can be obtained using completely unsupervised learning from the web, as performed by TEASE. However, the upper bound for our current implemented system is only about 50% because our syntactic matching does not handle morphological derivations.

6.3 System Results

experiment	recall	precision	F_1
input	0.18	0.62	0.28
input + iterative	0.29	0.42	0.34

Table 9: System results on the test set.

Table 9 presents our system results for the test set, corresponding to the first two experiments in Table 8. The recall achieved by our current implementation is notably worse than the upper bound of the manual analysis because of two general setbacks of the current syntactic matcher: 1) parsing errors; 2) limited transformation rule coverage.

First, the texts from the biology domain presented quite a challenge for the Minipar parser. For example, in the sentences containing the phrase '*X* bind specifically to *Y*' the parser consistently attaches the *PP* 'to' to 'specifically' instead of to 'bind'. Thus, the template '*X* bind to *Y*' cannot be directly matched.

Second, we manually created a small number of transformation rules that handle various syntactic phenomena, since we aimed at generic domain independent rules. The most difficult phenomenon to model with transformation rules is transparent heads. For example, in "*the dimerization of Prot1 interacts with Prot2*", the transparent head 'dimerization of *X*' is domain dependent. Transformation rules that handle such examples are difficult to acquire, unless a domain specific learning approach (either supervised or unsupervised) is used. Finally, we did not handle co-reference resolution in the current implementation.

Bunescu et al. (2005) and Bunescu and Mooney (2005) approached the protein interaction RE task using both handcrafted rules and several supervised Machine Learning techniques, which utilize about 180 manually annotated abstracts for training. Our results are not directly comparable with theirs because they adopted 10-fold cross-validation, while we had to divide the dataset into a development and a test set, but a rough comparison is possible. For the same 30% recall, the rule-based method achieved precision of 62% and the best supervised learning algorithm achieved precision of 73%. Comparing to these supervised and domain-specific rule-based approaches our system is noticeably weaker, yet provides useful results given that we supply very little domain specific information and acquire the paraphrasing templates in a fully unsupervised manner. Still, the matching models need considerable additional research in order to achieve the potential performance suggested by TEASE.

7 Conclusions and Future Work

We have presented a paraphrase-based approach for relation extraction (RE), and an implemented system, that rely solely on unsupervised paraphrase acquisition and generic syntactic template matching. Two targets were investigated: (a) a mostly unsupervised, domain independent, configuration for RE, and (b) an evaluation scheme for paraphrase acquisition, providing a first evaluation of its realistic coverage. Our approach differs from previous unsupervised IE methods in that we identify instances of a specific relation while prior methods identified template relevance only at the general scenario level.

We manually analyzed the potential of our approach on a dataset annotated with protein interactions. The analysis shows that 93% of the interacting protein pairs can be potentially identified with the template-based approach. Additionally, we manually assessed the coverage of the TEASE acquisition algorithm and found that 63% of the distinct pairs can be potentially recognized with the learned templates, assuming an ideal matcher, indicating a significant potential recall for completely unsupervised paraphrase acquisition. Finally, we evaluated our current system performance and found it weaker than supervised RE methods, being far from fulfilling the potential indicated in our manual analyses due to insufficient syntactic matching. But, even our current performance may be considered useful given the very small amount of domain-specific information used by the system.

Most importantly, we believe that our analysis and evaluation methodologies for an RE dataset provide an excellent benchmark for unsupervised learning of paraphrases and entailment rules. In the long run, we plan to develop and improve our acquisition and matching algorithms, in order to realize the observed potential of the paraphrase-based approach. Notably, our findings point to the need to learn generic morphological and syntactic variations in template matching, an area which has rarely been addressed till now.

Acknowledgements

This work was developed under the collaboration ITC-irst/University of Haifa. Lorenza Romano has been supported by the ONTOTEXT project, funded by the Autonomous Province of Trento under the FUP-2004 research program.

References

Razvan Bunescu and Raymond J. Mooney. 2005. Subsequence kernels for relation extraction. In *Proceedings of the 19th Conference on Neural Information Processing Systems*, Vancouver, British Columbia.

Razvan Bunescu, Ruifang Ge, Rohit J. Kate, Edward M. Marcotte, Raymond J. Mooney, Arun K. Ramani, and Yuk Wah Wong. 2005. Comparative experiments on learning information extractors for proteins and their interactions. *Artificial Intelligence in Medicine*, 33(2):139–155. Special Issue on Summarization and Information Extraction from Medical Documents.

Ido Dagan and Oren Glickman. 2004. Probabilistic textual entailment: Generic applied modeling of language variability. In *Proceedings of the PASCAL Workshop on Learning Methods for Text Understanding and Mining*, Grenoble, France.

Charles J. Fillmore, Collin F. Baker, and Hiroaki Sato. 2002. Seeing arguments through transparent structures. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC 2002)*, pages 787–791, Las Palmas, Spain.

Ralph Grishman, Lynette Hirschman, and Ngo Thanh Nhan. 1986. Discovery procedures for sublanguage selectional patterns: Initial experiments. *Computational Linguistics*, 12(3).

Takaaki Hasegawa, Satoshi Sekine, and Ralph Grishman. 2004. Discoverying relations among named entities from large corpora. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)*, Barcelona, Spain.

Dekang Lin and Patrick Pantel. 2001. Discovery of inference rules for question answering. *Natural Language Engineering*, 7(4):343–360.

Dekang Lin. 1998. Dependency-based evaluation on MINIPAR. In *Proceedings of LREC-98 Workshop on Evaluation of Parsing Systems*, Granada, Spain.

Catherine Macleod, Ralph Grishman, Adam Meyers, Leslie Barrett, and Ruth Reeves. Nomlex: A lexicon of nominalizations. In *Proceedings of the 8th International Congress of the European Association for Lexicography*, Liege, Belgium.

Deepak Ravichandran and Eduard Hovy. 2002. 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.

Yusuke Shinyama, Satoshi Sekine, Kiyoshi Sudo, and Ralph Grishman. 2002. Automatic paraphrase acquisition from news articles. In *Proceedings of the Human Language Technology Conference (HLT 2002)*, San Diego, CA.

Mark Stevenson and Mark A. Greenwood. 2005. A semantic approach to IE pattern induction. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, Ann Arbor, Michigan.

K. Sudo, S. Sekine, and R. Grishman. 2003. An improved extraction pattern representation model for automatic IE pattern acquisition. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL 2003)*, Sapporo, Japan.

Idan Szpektor, Hristo Tanev, Ido Dagan, and Bonaventura Coppola. 2004. Scaling web-based acquisition of entailment relations. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, Barcelona, Spain.

Roman Yangarber, Ralph Grishman, Pasi Tapanainen, and Silja Huttunen. 2000. Automatic acquisition of domain knowledge for information extraction. In *Proceedings of the 18th International Conference on Computational Linguistics*, Saarbruecken, Germany.