Learning Canonical Forms of Entailment Rules

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

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

Abstract

We propose a modular approach to paraphrase and entailment-rule learning that addresses the morpho-syntactic variability of lexical-syntactic templates. Using an entailment module that captures generic morpho-syntactic regularities, we transform every identified template into a canonical form. This way, statistics from different template variations are accumulated for a single template form. Additionally, morpho-syntactic redundant rules are not acquired. This scheme also yields more informative evaluation for the acquisition quality, since the bias towards rules with many frequent variations is avoided.

Keywords

Textual Entailment, Paraphrases, Knowledge Acquisition

1 Introduction

In many NLP applications such as Question Answering (QA) and Information Extraction (IE), it is crucial to recognize that a specific target meaning can be inferred from different text variants. For example, a QA system have to deduce that "*Mozart wrote the Jupiter symphony*" can be inferred from "*Mozart composed the Jupiter symphony*" in order to answer "*Who wrote the Jupiter symphony?*". This type of reasoning has been identified as a core semantic inference paradigm by the generic *textual entailment* framework [5].

An important type of knowledge representation needed for such inference is *entailment rules*. An entailment rule, e.g. 'X compose $Y \rightarrow X$ write Y', is a directional relation between two *templates*. Templates represents text patterns with variables that typically corresponds to semantic predicates. In an entailment rule, the left hand side template is assumed to entail the right hand side template in certain appropriate contexts, under the same variable instantiation. Such rules capture rudimentary inferences and are used as building blocks for more complex inference. For example, given the above entailment rule, a QA system can identify "*Mozart*" as the answer for the above question. A major obstacle for further advances in semantic inference is the lack of broad-scale knowledge-bases for such rules [1]. This need sparked intensive research on automatic acquisition of entailment rules (and similarly paraphrases). These algorithms' strength is in learning relations between lexical-syntactic templates, which capture lexical-based knowledge and world knowledge (see Section 2.1).

One noticeable phenomenon of lexical-syntactic templates is that they have many morpho-syntactic variations, which (largely) represent the same predicate and are semantically equivalent. For example, 'X compose Y' can be expressed also by 'Y is composed by X' or 'X's composition of Y'. Current learning algorithms ignore this morpho-syntactic variability. They treat these variations as semantically different, learning rules for each variation separately. This leads to several undesired consequences. First, statistics for a particular semantic predicate are scattered among different templates. This may result in insufficient statistics for learning a rule in any of its variations. Second, though rules may be learned in several variations (see Table 1), in most cases only a small part of the morpho-syntactic variations are learned. Thus, an inference system that uses only these learned rules would miss recognizing a substantial number of variations of the sought predicate.

It therefore makes more sense to design a modular architecture. In it, a separate entailment module recognizes entailing variations that are based on generic morphological and syntactic regularities (morpho-syntactic entailments). We propose to use such a module first at learning time, by learning only canonical forms of templates and rules. Then, applying the module also at inference time, in conjunction with the learned lexical-based canonical rules, guarantees the coverage of all morpho-syntactic variations of a given canonical rule.

Our proposed approach poses two advantages. First, the statistics from the different morpho-syntactic variations accumulate for one template form only. The improved statistics may result, for example, in learning more rules. Second, the learning output is without redundancies due to variations of the same predicate. Additionally, the evaluation of learning algorithms is more accurate when the bias towards templates with many frequent variations is avoided.

In this work we implemented a morpho-syntactic entailment module that utilizes syntactic rules for major syntactic phenomena (like passive and conjunctions) and morphological rules that address nominalizations. We then applied the module within two entailment rule acquisition algorithms. We measured redundancy removal of about 6% out of all rules learned. For one of the algorithms, we measured an increase of about 12% in the number of lexically different correct templates that were learned using our approach. Finally, we applied the morpho-syntactic entailment module also at inference time in a Relation Extraction setup for protein-interaction. In a preliminary experiment, we found that the rules learned using our new scheme yielded some improvement in recall.

Morpho-Syntactic Variations	
X compose Y $\rightarrow$ X write Y	X is composed by Y $\rightarrow$ X write Y
X accuse Y $\leftrightarrow$ X blame Y	X's accusation of Y $\leftrightarrow$ X blame Y
X acquire Y $\rightarrow$ X obtain Y	acquisition of Y by X $\rightarrow$ Y is obtained by X

Table 1: *Examples of learned rules that differ only in their morpho-syntactic structure.*

Template	Single-feature Approach (DIRT)		Anchor-Set Approach
	X-vector Features	Y-vector Features	Common Features
X compose Y	Bach, Beethoven Mozart, he	symphony, music sonata, opera	$\{X$='Mozart';Y='Jupiter symphony'$\}$, $\{X$='Bach';Y='Sonata Abassoonata'$\}$
X write Y	Tolstoy, Bach, author, Mozart, he	symphony, anthem, sonata, book, novel	

Table 2: *Examples for features of the anchor set and single-feature approaches for two related templates.*

2 Background

2.1 Entailment Rule Learning

Many algorithms for automatically learning entailment rules and paraphrases (which can be viewed as bidirectional entailment rules) were proposed in recent years. These methods recognize templates in texts and identify entailment relations between them based on shared features.

These algorithms may be divided into two types. The prominent approach identify an entailment relation between two templates by finding variable instantiation tuples, termed here *anchor-sets*, that are common to both templates [13, 18, 2, 12, 20, 16]. Anchor-sets are complex features, consisting of several terms, labelled by their corresponding variables. Table 2 (right column) presents common anchor-sets for the related templates 'X compose Y' and 'X write Y'. Typically, only few common anchor-sets are identified for each entailment relation.

A different single-feature approach is proposed by the DIRT algorithm [10]. It uses simple, less informative but more frequent features. It constructs a feature vector for each variable of a given template, representing the context words that fill the variable in the different occurrences of the template in the corpus. Two templates are identified as semantically related if they have similar vectors. Table 2 shows examples for features of this type. DIRT parses a whole corpus and limits the allowed structures of templates only to paths in the parse graphs, connecting nouns at their ends.

In this paper we implemented the TEASE algorithm [20]. It is an unsupervised algorithm that acquires entailment relations from the Web for given input templates using the anchor-set approach (we required at least two common anchor-sets for learning a relation). We also implemented the DIRT algorithm over a local corpus, the first CD of Reuters RCV1[1]. Both algorithms process *lexical-syntactic templates*, which are represented by parse subtrees. All sentences are parsed using the Minipar dependency parser [9].

For a given input template I, these algorithms can be viewed as learning a list of output templates $\{O_j\}_1^{n_I}$, where n_I is the number of templates learned for I. Each output template is suggested as holding an entailment relation with the input template, but current algorithms do not specify the entailment direction(s). Thus, each pair $\{I, O_j\}$ induces two candidate directional entailment rules: '$I \rightarrow O_j$' and '$O_j \rightarrow I$'.

As shown in previous evaluations the precision of DIRT and TEASE is limited [10, 2, 20, 19]. Currently, their application should typically involve manual filtering of the learned rules, and the algorithms' utility is reflected mainly by the amount of correct rules they learn. Specifically, DIRT learns a long tail of low quality rules with less significant statistics, which still yield a positive similarity value.

The learned entailment rules and paraphrases can be used at *inference time* in applications such as IE [18, 14, 17] and QA [10, 13, 8], where matched rules deduce new target predicate instances from texts (like the 'compose → write' example in Section 1).

2.2 Morpho-Syntactic Template Variations

Lexical syntactic templates can take on many morpho-syntactic variations, which are usually semantically equivalent. This phenomenon is addressed at the inference phase by recognizing semantically equivalent syntactic variations, such as passive forms and conjunctions (e.g. [14]). Some work was done to systematically recognize morphological variations of predicates [11, 7], but it was not applied for entailment inference.

In contrast, current methods for learning lexical-syntactic rules do not address the morpho-syntactic variability at learning time at all. Thus, they learn rules separately for each variation. This results in either learning redundant rules (see Table 1) or missing some of the relevant rules that occur in a corpus. Moreover, some rules might not be learned in any variation. For example, if for each of the rules 'X acquire Y $\rightarrow$ X own Y', 'Y is acquired by X $\rightarrow$ X own Y' and 'X's acquisition of Y $\rightarrow$ X own Y' there are no sufficient statistics then none of them will be learned.

To sum up, though several problems rise from disregarding the morpho-syntactic variability, there is still no sound solution for addressing it at learning time.

[1] http://about.reuters.com/researchandstandards/corpus/

3 A Modular Approach for Entailment Rule Learning

A natural solution for addressing the morpho-syntactic variability in templates is a modular architecture, in which a separate entailment module recognizes entailing variations that are based on generic morphological and syntactic regularities.

In our scheme, we use this morpho-syntactic entailment module to transform lexical-syntactic template variations that occur in a text into their *canonical form*. This form, which we chose to be the active verb form with direct modifiers, is entailed by other template variations. We next describe our implementation of such a module and its application within entailment rule acquisition algorithms.

3.1 Morpho-Syntactic Canonization Module

We implementated a morpho-syntactic module based on a set of *canonization rules*, highly accurate morpho-syntactic entailment rules. Each rule represents one morpho-syntactic regularity that is eliminated when the rule is applied to a given template (see examples in Table 3 and Figure 1).

Our current canonization rule collection consists of two types of rules: (a) syntactic-based rules; (b) morpho-syntactic nominalization rules. We next describe each rule type. As we use the Minipar parser, all rules are adapted to Minipar's output format.

Syntactic-based Rules These rules capture entailment patterns associated with common syntactic structures. Their function is to simplify and generalize the syntactic structure of a template.

In the current implementation we manually created the following simplification rules: (a) passive forms into active forms; (b) removal of conjunctions; (c) removal of appositions; (d) removal of abbreviations; (e) removal of set description by the 'such as' preposition. Table 3 presents some of the rules we created together with examples of their effect.

Nominalization Rules Entailment rules such as 'acquisition of Y by $X \rightarrow X$ acquire Y' and 'Y's acquisition by $X \rightarrow X$ acquire Y' capture the relations between verbs and their nominalizations. We automatically derived these rules from Nomlex, a hand-coded database of about 1000 English nominalizations [11], as described in [15]. These rules transform any nominal template in Nomlex into its related verbal form. These rules preserve the semantics of the original template predicate. We chose the verbal form as the canonical form since for every predicate with specific semantic modifiers there is only one verbal active form in Nomlex, but typically several equivalent nominal forms.

Chaining of Canonization Rules Each of the syntactic rules decreases the size of a template. In addition, nominalization rules can only be applied once for a given template, since no rule in our rule-set transforms a verbal template into one of its nominal forms. Thus, applying rules until no rule can apply is a finite process. In addition, each of our rules is independent of the others, operating on a different set of dependency relations. Consequently, applying any sequence of rules until no other rule can apply will result in the same final canonical template form. Figure 1 illustrates an example for rule chaining.

3.2 Applying the Canonization Module

When a morpho-syntactic entailment module is utilized at inference time (e.g. [14]), it recognizes a closure of morpho-syntactic variations for a lexical-syntactic template. Accordingly, acquisition algorithms may learn just a single morpho-syntactic variation of a template.

With this modular scheme in mind, we propose to solve the learning problems discussed in Section 2.2 by utilizing the morpho-syntactic entailment module at learning time as well. We incorporate the module in the learning algorithms (TEASE and DIRT in our experiment) by converting each template variation occurrence in the learning corpus into an occurrence of a canonical template. Thus, the learning algorithms operate only on canonical forms.

As discussed in Section 1, when canonization is used, no morpho-syntactically redundant rules are learned, with respect to the variations that are recognized by the module. This makes the output more compact, both for storage and for use. In addition, the statistical reliability of learned rules may be improved. For example, rules that could not be learned before in any variation may be learned now for the canonical form.

Methodologically, previous evaluations of learning algorithms reported accuracy relative to the redundant list of rules, which creates a bias for templates with many frequent variations. When this bias is removed and only truly different lexical-syntactic rules are assessed, evaluation is more efficient and accurate.

4 Evaluation

We conducted two experiments: (a) a manual evaluation of the contribution of the canonization module to TEASE and DIRT, based on human judgment of the learned rules; (b) a Relation Extraction evaluation setup for a protein interaction data-set.

4.1 Human Judgement Evaluation

We have selected 20 different verbs and verbal phrases[2] as input templates for both TEASE and DIRT, and executed both the baseline versions (without canonization), marked as $TEASE_b$ and $DIRT_b$, and the versions with the canonization module, marked as $TEASE_c$ and $DIRT_c$. The results of the executions constitute our test-set rules.

As discussed in Section 2.1, both TEASE and DIRT do not learn the direction(s) of an entailment relation between an input template I and a learned output template O. Thus, we evaluated both candidate directional rules, '$I \rightarrow O$' and '$O \rightarrow I$'.

Rule Evaluation The prominent approach for evaluating rules is to present them to human judges, who assess whether each rule is correct or not. Generally, a rule is considered correct if the judge could think of reasonable

[2] The verbs are: accuse, approve, calculate, change, demand, establish, finish, hit, invent, kill, know, leave, merge with, name as, quote, recover, reflect, tell, worsen, write.

Rule	Description	Original Template	Simplified Template
passive to active	$X \xleftarrow{pcomp-n} by \xleftarrow{by-subj} V$ $\Longrightarrow X \xleftarrow{subj} V$	$X \xleftarrow{pcomp-n} by \xleftarrow{by-subj} find \xrightarrow{obj} Y$	$X \xleftarrow{subj} find \xrightarrow{obj} Y$
conjunction	$Z \xrightarrow{conj} Y \Longrightarrow Y$	$X \xleftarrow{subj} find \xrightarrow{obj} gold \xrightarrow{conj} Y$	$X \xleftarrow{subj} find \xrightarrow{obj} Y$
apposition	$Z \xrightarrow{appo} Y \Longrightarrow Y$	$X \xleftarrow{subj} find \xrightarrow{obj} protein \xrightarrow{appo} Y$	$X \xleftarrow{subj} find \xrightarrow{obj} Y$
abbreviation	$Z \xrightarrow{spellout} Y \Longrightarrow Y$	$X \xleftarrow{subj} find \xrightarrow{obj} NDA \xrightarrow{spellout} Y$	$X \xleftarrow{subj} find \xrightarrow{obj} Y$

Table 3: *Some of the syntactic rules used in our implementation, together with usage examples (the application of the second rule and the third rule is demonstrated in Figure 1).*

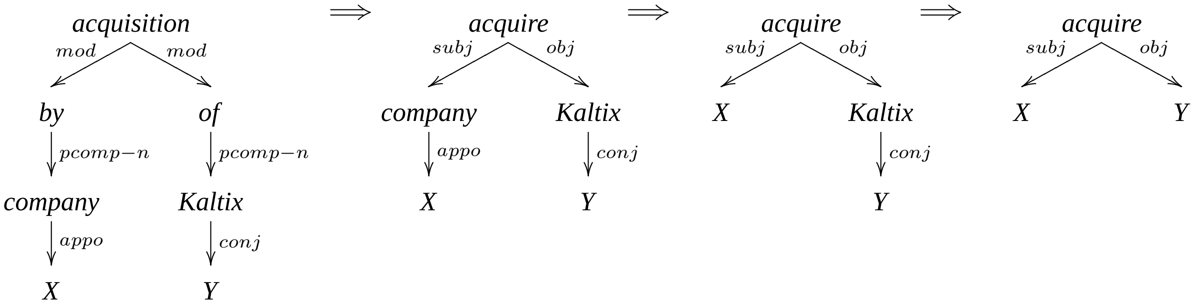

Fig. 1: *Chaining of canonization rules that transforms the path template between the arguments* $\{X=$*'Google'*$;Y=$*'Sprinks'*$\}$*, which occurs in the sentence* "We witnessed the acquisition of Kaltix and Sprinks by another growing company, Google"*, into a canonized template form. The first rule applied is a nominalization rule, followed by removal of apposition and removal of conjunction (as described in Table 3). As can be seen, applying the rules in any order will result in the same final canonized form.*

contexts under which it holds. However, it is difficult to explicitly define when a learned rule should be considered correct under this methodology.

Instead, we follow the evaluation methodology presented in [19], where each rule '$L \rightarrow R$' is evaluated by presenting the judges not only with the rule but rather with a sample of sentences that match its left hand side L. The judges then assess whether the rule holds under each specific example sentence. The precision of a rule is computed by the percentage of examples for which entailment holds out of all "relevant" examples in the judged sample. A rule is considered correct if its precision is higher than 0.8 (see [19] for details). This instance-based approach was shown to be more reliable than the rule-based approach.

4.2 TEASE Evaluation

We separated the templates that were learned by $TEASE_c$ into two lists: (a) a *baseline-templates* list containing templates also learned by $TEASE_b$; (b) a *new-templates* list containing templates that were not learned by $TEASE_b$, but learned by $TEASE_c$ thanks to the improved statistics. In total, 3871 templates were learned: 3309 in the baseline-templates list and 562 in the new-templates list. Inherently, every output template learned by $TEASE_b$ is also learned in its canonical form by $TEASE_c$, since its supporting statistics may only increase.

We randomly sampled 100 templates from each list and evaluated their correctness according to the methodology in Section 4.1. We retrieved 10 example sentences for each rule from the first CD of Reuters RCV1. Two judges, fluent English speakers, evaluated the examples. We randomly split the rules between the judges with 100 rules (942 examples) cross annotated for agreement measurement.

Results First, we measured the redundancy in the rules learned by $TEASE_b$ to be 6.2% per input template on average. We considered only morpho-syntactic phenomena that are addressed in our implementation. This redundancy was eliminated using the canonization module.

Next, we evaluated the quality of each rule sampled using two scores: (1) micro average ***Precision***, the percentage of correct templates out of all learned templates, and (2) average ***Yield***, the average number of correct templates learned for each input template, as extrapolated for the sample. The results are presented in Table 5. The agreement between the judges was measured by the Kappa value [4], which is 0.67 on the relevant examples (corresponding to substantial agreement).

We expect $TEASE_c$ to learn new rules using the canonization module. In our experiment, 5.8 more correct templates were learned on average per input template by $TEASE_c$. This corresponds to an increase of 11.6% in average Yield (see Table 5). Examples of new correctly

Rule	Sentence	Judgment
X clarify $Y \rightarrow X$ prepare Y	**He** didn't clarify **his position on the subject**.	Left not entailed
X hit $Y \rightarrow X$ approach Y	**Other earthquakes** have hit **Lebanon** since '82.	Irrelevant context
X regulate $Y \rightarrow X$ reform Y	**The SRA** regulates **the sale of sugar**.	No entailment
X stress $Y \rightarrow X$ state Y	**Ben Yahia** also stressed **the need for action**.	Entailment holds

Table 4: *Example sentences for rules and their evaluation judgment.*

learned templates are shown in Table 6.

There is a slight decrease in precision when using $TEASE_c$. One possible reason is that the new templates are usually learned from very few occurrences of different variations, accumulated for the canonical templates. Thus, they may have a somewhat lower precision in general. Overall, the significant increase in Yield is much more important, especially if the learned rules are later filtered manually (see Section 2.1).

Template List	Avg. Precision	Avg. Yield
$TEASE_b$	30.1%	49.8
$TEASE_c$	28.7%	55.6
$DIRT_b$	24.7%	46.9
$DIRT_c$	24.9%	47.5

Table 5: *Average Precision and Yield of the output lists.*

Input Template	Learned Template
X accuse Y	X blame Y
X approve Y	X take action on Y
X demand Y	X call for Y, X in demand for Y
X establish Y	X open Y
X hit Y	X slap Y
X invent Y	grant X patent on Y, X is co-inventor of Y
X kill Y	X hang Y, charge X in death of Y
X named as Y	hire X as Y, select X as Y
X quote Y	X cite Y
X tell Y	X persuade Y, X say to Y
X worsen Y	X impair Y

Table 6: *Examples for correct templates that TEASE learned only after using canonization rules.*

4.3 DIRT Evaluation

Unlike TEASE, DIRT has a very long noisy tail of candidate templates (see Section 2.1). However, DIRT poses no hard threshold for filtering out this long tail. Instead, we follow [10], who evaluated only the top-N templates learned for each input template. [10] set N to be 40, but this choice seems quite arbitrary. We set N to be 190 to assess an output list that is similar in size to TEASE's output. Before selecting the top 190 templates, we removed redundant templates from $DIRT_b$, those that are just morpho-syntactic variations of a template with a higher score. We converted the remaining templates to their canonical forms.

We separated the templates learned for each input template into three lists: (a) a *common-templates* list containing templates that appear in both $DIRT_b$ and $DIRT_c$ top-190 lists; (b) a *new-templates* list containing templates that appear only in the $DIRT_c$ list; (c) an *old-templates* list containing templates that appear only in the $DIRT_b$ list. Out of the 3800 templates selected from each DIRT version output, 3353 were in the common-list and 447 were in each of the new/old lists.

We sampled 100 templates from each list and evaluated their correctness (10 sentences for each rule). One judge evaluated the sample. The evaluation results were affirmed by an additional evaluation by one of the authors.

Results We measured the redundancy in the rules learned by $DIRT_b$ to be 5.6% per input template on average. This redundancy was removed using the canonization module. We found that only about 13% of the learned templates were learned by both TEASE and DIRT. This shows that the algorithms do not compete but rather largely complement each other in terms of Yield, since they learn from different resources.

13.3% of the top-190 templates learned by $DIRT_b$ were replaced by other templates in $DIRT_c$, as the change in statistics results in different template ranking. We measured Precision and Yield as in Section 4.2. The results are presented in Table 5.

As can be seen, the performance of $DIRT_c$ is basically comparable to that of $DIRT_b$. It seems that in typical paraphrase acquisition algorithms like TEASE, which use complex and more informative features that are infrequent, adding more statistics results in higher quality learning. On the other hand, DIRT is based on frequent simple features that are less informative. Under this approach, adding some more statistics does not seem to dramatically change the overall score of a rule. Perhaps a more substantial increase in the statistics, such as by adding more canonization rules, will result in a positive change.

Overall, it is useful to incorporate canonization also in DIRT in order to remove the redundancy within the learned rules but also to enable a uniform architecture for applying rules learned by different algorithms.

4.4 Relation Extraction Evaluation

To illustrate the potential contribution of the increased number of learned rules we conducted a small-scale experiment in a Relation Extraction (RE) setup over a data-set of protein interactions [3]. The task is to identify pairs of proteins that are described in a text as interacting.

We have set a simple partial replication of the RE configuration presented in [14]. We used 'X interact with Y' as the only input template for both $TEASE_b$ and $TEASE_c$, which learned entailment rules containing this template from the Web. We then extracted protein pairs using the rules learned. For canonization at inference time. we used only the rules described in Section 3.1 (a wider range of matching techniques should be used in order to reach higher recall).

Table 7 presents the results of our two TEASE versions for a test set of about 600 mentions of interacting pairs. There is a relative improvement of about 10% in recall, which reflects the yield increase in $TEASE_c$. These results are preliminary and of small scale, but they illustrate the potential benefit of learning with canonization.

We note that TEASE precision in this experiment, which was measured over actual applications of the learned rules in the test set, is much higher than that of Section 4.2, where the percentage of correctly learned rules was measured. This shows that many incorrectly learned rules are not applicable in typical contexts and thus rarely deteriorate overall performance.

Implementation	Recall	Precision
$TEASE_b$	9.4%	83%
$TEASE_c$	10.4%	87.5%

Table 7: *Results for the protein interaction setup using TEASE with and without canonization.*

4.5 Analysis

Parser errors are one of the main reasons that variations are sometimes not transformed into their canonical form. These errors result in different parse trees for the same syntactic constructs. Thus, several parser dependent rules may be needed to capture the same phenomenon. Moreover, it is difficult to design canonization rules for some parsing errors, since the resulting parse trees consist of structures that are common to other irrelevant templates. For example, when Minipar chooses the head of the conjunct '*Y*' in "*The interaction between X and Y will not hold for long*" to be '*interaction*' and not '*X*', the appropriate nominalization rule cannot be applied. These errors affect both the learning phase, where statistics are not accumulated to the appropriate canonical form, and the inference phase, where a variations of a canonical rule are not recognized.

Finally, we note that the reported results correspond only to the phenomena captured by our currently implemented canonization rules. Adding more rules that cover more morpho-syntactic phenomena is expected to increase the performance obtained by our canonization scheme. For example, there are many nominalizations that are not specified in the current Nomlex version, but can be found in other resources, such as WordNet [6].

5 Conclusions

We proposed a modular approach for addressing morpho-syntactic variations of templates when learning entailment rules, based on rule canonization. We then used it for template canonization in two state-of-the-art acquisition algorithms. Our experiments showed that redundancy is removed while new correct rules are learned. We also showed initial improvement in a Relation Extraction setting when using the additional rules learned with the canonization module. Finally, we suggest that the evaluation of rules in a canonical form is more accurate, since the bias for templates with many frequent variations learned is removed.

In future work we plan to investigate other types of entailment knowledge that can contribute to canonization, such as synonyms. We also plan to add additional syntactic and morpho-syntactic rules, which were not covered yet.

Acknowledgements

The authors would like to thank Chen Erez for her help in the experiments. We also want to thank Efrat Brown, Ruthie Mandel and Malky Rabinowitz for their evaluation. This work was partially supported by the Israeli Ministry of Industry and Trade under NEGEV Consortium (www.negev-initiative.org) and the IST Programme of the European Community under the PASCAL Network of Excellence IST-2002-506778.

References

[1] R. Bar-Haim, I. Dagan, B. Dolan, L. Ferro, D. Giampiccolo, B. Magnini, and I. Szpektor. The second pascal recognising textual entailment challenge. In *Second PASCAL Challenge Workshop for Recognizing Textual Entailment*, 2006.

[2] R. Barzilay and L. Lee. Learning to paraphrase: An unsupervised approach using multiple-sequence alignment. In *Proceedings of NAACL-HLT*, 2003.

[3] R. Bunescu, R. Ge, K. J. Rohit, E. M. Marcotte, R. J. Mooney, A. K. Ramani, and Y. W. Wong. Comparative experiments on learning information extractors for proteins and their interactions. *Artificial Intelligence in Medicine (Special Issue on Summarization and Information Extraction from Medical Documents)*, 2004.

[4] J. Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46, 1960.

[5] I. Dagan, O. Glickman, and B. Magnini. The pascal recognising textual entailment challenge. *Lecture Notes in Computer Science*, 3944:177–190, 2006.

[6] C. Fellbaum, editor. *WordNet: An Electronic Lexical Database*. Language, Speech and Communication. MIT Press, 1998.

[7] O. Gurevich, R. S. Crouch, T. H. King, and V. de Paiva. Deverbal nouns in knowledge representation. In *Proceedings of FLAIRS*, 2006.

[8] S. Harabagiu and A. Hickl. Methods for using textual entailment in open-domain question answering. In *Proceedings of ACL*, 2006.

[9] D. Lin. Dependency-based evaluation of minipar. In *Proceedings of the Workshop on Evaluation of Parsing Systems at LREC*, 1998.

[10] D. Lin and P. Pantel. Discovery of inference rules for question answering. *Natural Language Engineering*, 7(4):343–360, 2001.

[11] C. Macleod, R. Grishman, A. Meyers, L. Barrett, and R. Reeves. Nomlex: A lexicon of nominalizations. *Proceedings of EURALEX*, 1998.

[12] C. Quirk, C. Brockett, and W. Dolan. Monolingual machine translation for paraphrase generation. In *Proceedings of EMNLP*, 2004.

[13] D. Ravichandran and E. Hovy. Learning surface text patterns for a question answering system. In *Proceedings of ACL*, 2002.

[14] L. Romano, M. Kouylekov, I. Szpektor, I. Dagan, and A. Lavelli. Investigating a generic paraphrase-based approach for relation extraction. In *Proceedings of EACL*, 2006.

[15] T. Ron. Generating entailment rules using online lexical resources. *Mastersś thesis, Computer Science Department, Bar Ilan University*, 2006.

[16] S. Sekine. Automatic paraphrase discovery based on context and keywords between ne pairs. In *Proceedings of IWP*, 2005.

[17] S. Sekine. On-demand information extraction. In *Proceedings of the COLING/ACL Main Conference Poster Sessions*, 2006.

[18] Y. Shinyama, S. Sekine, K. Sudo, and R. Grishman. Automatic paraphrase acquisition from news articles. In *Proceedings of HLT*, 2002.

[19] I. Szpektor, E. Shnarch, and I. Dagan. Instance-based evaluation of entailment rule acquisition. In *Proceedings of ACL*, 2007.

[20] I. Szpektor, H. Tanev, I. Dagan, and B. Coppola. Scaling web-based acquisition of entailment relations. In *Proceedings of EMNLP*, 2004.