# Inference Rules for Recognizing Textual Entailment

Georgiana Dinu
Saarland University
dinu@coli.uni-sb.de

Rui Wang
Saarland University
rwang@coli.uni-sb.de

**Abstract**

In this paper, we explore the application of inference rules for recognizing textual entailment (RTE). We start with an automatically acquired collection and then propose methods to refine it and obtain more rules using a hand-crafted lexical resource. Following this, we derive a dependency-based representation from texts, which aims to provide a proper base for the inference rule application. The evaluation of our approach on the RTE data shows promising results on precision and the error analysis suggests future improvements.

## 1 Introduction

Textual inference plays an important role in many natural language processing (NLP) tasks, such as question answering [Harabagiu and Hickl, 2006]. In recent years, the recognizing textual entailment (RTE) [Dagan et al., 2006] challenge, which focuses on detecting semantic inference, has attracted a lot of attention. Given a text **T** (several sentences) and a hypothesis **H** (one sentence), the goal is to detect if **H** can be inferred from **T**.

Studies such as [Clark et al., 2007] attest that lexical substitution (e.g. synonyms, antonyms) or simple syntactic variation accounts for the entailment only in a small number of pairs. Thus, one essential issue is to identify more complex expressions which, in appropriate contexts, convey the same (or similar) meaning. More generally, we are also interested in pairs of expressions in which only a uni-directional inference relation holds[1].

A typical example is the following RTE pair in which *accelerate to* in **H** is used as an alternative formulation for *reach speed of* in **T**.

[1]We will use the term inference rule to stand for such concept; the two expressions can be actual paraphrases if the relation is bi-directional

**T:** *The high-speed train, scheduled for a trial run on Tuesday, is able to* ***reach*** *a maximum* ***speed of*** *up to 430 kilometers per hour, or 119 meters per second.*

**H:** *The train* ***accelerates*** *to 430 kilometers per hour.*

One way to deal with textual inference is through rule representation, such as *X wrote Y* $\approx$ *X is author of Y*. However, manually building collections of inference rules is time-consuming and it is unlikely that humans can exhaustively enumerate all the rules encoding the knowledge needed in reasoning with natural languages. Instead, an alternative is to acquire these rules automatically from large corpora. Furthermore, given such a rule collection, how to successfully use it in NLP applications is the next step to be focused on.

For the first aspect, we extend and refine an existing collection of inference rules acquired based on the *Distributional Hypothesis* (DH). One of the main advantages of using DH is that the only input needed is a large corpus of (parsed) text[2]. For this purpose, a hand-crafted lexical resource is used for augmenting the original inference rule collection and excluding some of the incorrect rules.

For the second aspect, we focus on applying these rules to the RTE task. In particular, we use a structure representation derived from the dependency parse trees of **T** and **H**, which aims to capture the essential information they convey.

The rest of the paper is organized as follows: Section 2 introduces the inference rule collection we use, based on the Discovery of Inference Rules from Text (henceforth DIRT) algorithm; we also discuss previous work on applying it to the RTE task. Section 3 presents our analyses on the RTE data and discusses two issues: the lack of rules and the difficulty of finding proper ways of applying them. Section 4 proposes methods to extend and refine the rule collection aiming at the former issue. To address the latter issue, Section 5 describes the structure representation we use to identify the appropriate context for the rule application. The experiments will be presented in Section 6, followed by an error analysis and discussions in Section 7. Finally, Section 8 will conclude the paper and point out some future work.

## 2 Background

A number of automatically acquired inference rule/paraphrase collections are available, such as [Szpektor et al., 2004]. In our work we use the DIRT collection because it is the largest one and it has a relatively good accuracy (in the 50% range, [Szpektor et al., 2007]). In this section, we describe the DIRT algorithm for ac-

[2]Another line of work on acquiring paraphrases uses comparable corpora, for instance [Barzilay and McKeown, 2001], [Pang et al., 2003]

quiring inference rules. Following that, we will overview the RTE systems which take DIRT as an external knowledge resource.

## 2.1 Discovery of Inference Rules from Text

The DIRT algorithm has been introduced by [Lin and Pantel, 2001] and it is based on what is called the *Extended Distributional Hypothesis*. The original DH states that *words* occurring in similar contexts have similar meaning, whereas the extended version hypothesizes that *phrases* occurring in similar contexts are similar.

An inference rule in DIRT is a pair of binary relations $\langle\ pattern_1(X, Y), pattern_2(X, Y)\ \rangle$ which stand in an inference relation. $pattern_1$ and $pattern_2$ are chains in Minipar [Lin, 1998] dependency trees while X and Y are placeholders for nouns at the end of the chains. The two patterns will constitute a candidate paraphrase if the sets of X and Y values exhibit relevant overlap. An example is the pair ($\mathbf{X} \xleftarrow{subj} prevent \xrightarrow{obj} \mathbf{Y}$, $\mathbf{X} \xleftarrow{subj} provide \xrightarrow{obj} protection \xrightarrow{mod} against \xrightarrow{pcomp} \mathbf{Y}$).

Such rules can be defined [Szpektor et al., 2007] as directional relations between two text patterns with variables. The left-hand-side pattern is assumed to entail the right-hand-side pattern in certain contexts, under the same variable instantiation. The definition relaxes the intuition of inference, as we only require the entailment to hold in $some$ but not $all$ contexts, motivated by the fact that such inferences occur often in natural text.

## 2.2 Related Work

Intuitively such inference rules should be effective for recognizing textual entailment. However, only a small number of systems have used DIRT as a resource in the RTE-3 challenge, and the experimental results have not shown its great contribution.

In [Clark et al., 2007]'s approach, semantic parsing in clause representation is performed and true entailment is decided only if every clause in the semantic representation of **T** semantically matches some clause in **H**. The only variation allowed consists of rewritings derived from WordNet and DIRT. Given the preliminary stage of this system, the overall results show very low improvement over a random classification baseline.

[Bar-Haim et al., 2007] implement a proof system using rules for generic linguistic structures, lexical-based rules, and lexical-syntactic rules (which were obtained with the DIRT algorithm applied to the first CD of the Reuters RCV1 corpus). Given a premise $p$ and a hypothesis $h$, the lexical-syntactic component marks all lexical noun alignments. For every pair of alignments, the paths between the

two nouns are extracted, and the DIRT algorithm is applied to obtain a similarity score. If the score is above a threshold, the rule will be applied. However, these lexical-syntactic rules are only used in about 3% of the attempted proofs and for most cases there is no lexical variation.

[Iftene and Balahur-Dobrescu, 2007] use DIRT in a more relaxed manner. A DIRT rule is employed in the system if at least one of the anchors match in **T** and **H**, i.e. they use them as unary rules. However, the analysis of the system shows that the DIRT component is the least relevant one (adding 0.4% to the precision).

In [Marsi et al., 2007]'s system, a paraphrase substitution step is added on top of a system based on a tree alignment algorithm. The basic paraphrase substitution method follows several steps. Initially, the two patterns of a rule are matched in **T** and **H** (instantiations of the anchors $X$, $Y$ do not have to match). The **T** tree is transformed by applying the paraphrase substitution. Following that, the transformed **T** tree and **H** tree are aligned. The coverage (proportion of aligned content words) is computed and if above some threshold, the entailment holds. The paraphrase component adds 1.0% to the result on the development set and only 0.5% to the test set, but a more detailed analysis on the interaction of this component with other components of the system is not given.

# 3 Inference Rules for RTE

In this section our goal is to investigate the causes for which a resource such as DIRT fails to bring clear improvements to RTE. The issues we have encountered can be divided into two categories. Firstly, given a collection of correct inference rules, making full use of the knowledge encoded in it is not a trivial task. Secondly, some of the needed rules still lack even in a very large collection such as DIRT. Section 4 will tackle the latter issue first while Section 5 will focus on the former one.

## 3.1 DIRT Rules Found in the RTE Data

To Address this first issue, we begin with a straightforward experiment to discover the number of pairs in the RTE data which contain rules present in DIRT [3].

Following the definition of an entailment rule, we identify RTE pairs in which $pattern_1(w1, w2)$ and $pattern_2(w1, w2)$ are matched, one in **T** and the other one in **H**, and thus, $\langle pattern_1(X, Y), pattern2(X, Y)\rangle$ is an inference rule. The pair

[3] For all the experiments in this paper, we use the DIRT collection provided by [Lin and Pantel, 2001], derived from the DIRT algorithm applied on 1GB of newstext.

below is an example of this.

**T:** *The sale was made to pay Yukos US$ 27.5 billion tax bill, Yuganskneftegaz was originally sold for US$ 9.4 billion to a little known company* ***Baikalfinansgroup*** *which was later* ***bought by*** *the Russian state-owned oil company* ***Rosneft.***

**H:** ***Baikalfinansgroup*** *was* ***sold to Rosneft.***

On average, only 2% of the pairs in the RTE data are subject to such inference rules. Out of these, approximately 50% are lexical rules (one verb entailing the other) and in the rest, around 50% are present in WordNet as a synonym, hypernym or sister relation.

However, given the small number of inference rules identified this way, we performed another analysis. This aims at determining an upper bound of the number of pairs featuring entailment phrases present in a collection. Given DIRT and the RTE data, we compute that in how many pairs two patterns of a paraphrase can be matched irrespectively of their anchor values. An example is the following pair,

**T:** *Libyas case against Britain and the US* ***concerns*** *the dispute over their demand for extradition of Libyans charged with blowing up a Pan Am jet over Lockerbie in 1988.*

**H:** *One case* ***involved*** *the extradition of Libyan suspects in the Pan Am Lockerbie bombing.*

This is a case in which the rule is correct and the entailment is positive. In order to determine this, a system will have to know that *Libya's case against Britain and the US* in **T** entails *one case* in **H**. Similarly, in this context, *the dispute over their demand for extradition of Libyans charged with blowing up a Pan Am jet over Lockerbie* can be replaced with *the extradition of Libyan suspects in the Pan Am Lockerbie bombing*. Altogether in around 25% of the pairs, patterns of a rule can be found in this way, and many times more than one rule in a pair. However, in many of these pairs, finding out the patterns of an inference rule does not imply that the rule is truly present in that pair.

Making use of the knowledge encoded with such rules is therefore, not a trivial task. If rules are used strictly in concordance with their definition, their utility is limited to a very small number of pairs. For this reason, 1) instead of forcing the anchor values to be identical as most previous works, we allow flexible rule matching (similar to [Marsi et al., 2007]) and 2) furthermore, we control the rule application process using a structure representation derived from the dependency tree (Section 5).

| | |
|---|---|
| *X, founded in Y* | $\rightarrow$ *X, opened in Y* |
| *X launch Y* | $\rightarrow$ *X produce Y* |
| *X represent Z* | $\rightarrow$ *X work for Y* |
| *X faces menace from Y* | $\leftrightarrow$ *X endangered by Y* |
| *X, peace agreement for Y* | $\rightarrow$ *X is formulated to end war in Y* |

Table 1: Example of inference rules needed in RTE

| *X face threat of Y* | *X at risk of Y* |
|---|---|
| *face* $\approx$ *confront, front, look, face up*<br>*threat* $\approx$ *menace, terror, scourge*<br>*endangerment, peril* | *risk* $\approx$ *danger, hazard, jeopardy* |

Table 2: Lexical variations creating new rules based on DIRT rule *X face threat of Y* $\rightarrow$ *X at risk of Y*

### 3.2 Missing Rules

Apart from the issues underlined in the previous section, looking at the data, we find it quite clear that DIRT lacks rules that many entailment pairs require.

Table 1 gives a selection of rules that are needed in some entailment pairs. The first three rows contain rules which are not structurally complex. These, however, are missing from both DIRT and also other hand-crafted resources such as WordNet (i.e. there is no short path connecting them). This is to be expected as they are rules which hold in some specific contexts, but difficult to be captured by a sense distinction of the lexical items involved. The more complex rules are even more difficult to be captured by a DIRT-like algorithm. Some of these do not occur frequently enough even in large amounts of text to permit the acquirement of them via DH.

## 4 Extending and Refining DIRT

In order to address the issue of missing rules, we investigate the effects of combining DIRT with an exact hand-coded lexical resource in order to create new rules.

For this we extended the DIRT rules by adding rules in which any of the lexical items involved in the patterns can be replaced by WordNet synonyms. The idea behind this is that a combination of various lexical resources is needed in order to cover the vast variety of phrases which humans can judge to be in an inference relation.

In the example above, we consider the DIRT rule *X face threat of Y* $\rightarrow$ *X, at risk*

*of Y* (Table 2). Of course at this moment due to the lack of sense disambiguation, our method introduces lots of rules that are not correct. As one can see, expressions such as *front scourge* do not make any sense, therefore any rules containing this will be incorrect. However some of the new rules created in this example, such as *X face threat of Y* $\approx$ *X, at danger of Y* are reasonable ones and the rules which are incorrect often contain patterns that are very unlikely to occur in natural text.

The method just described allows us to identify the first three rules listed in Table 1. We also acquire the rule *X face menace of Y* $\approx$ *X endangered by Y* (via *X face threat of Y* $\approx$ *X threatened by Y*, $menace \approx threat$, $threaten \approx endanger$). However the entailment pair requires a slightly different version of the rule, involving the phrase *face menace* ***from***.

Our extension is application-oriented therefore it is not intended to be evaluated as an independent rule collection, but in an application scenario such as RTE (Section 6).

Another issue that we address is the one of removing the most systematic errors present in DIRT. DH algorithms have the main disadvantage that not only phrases with the same meaning are extracted but also phrases with opposite meaning.

In order to overcome this problem and since such errors are relatively easy to detect, we applied a filter to the DIRT rules. This eliminates inference rules which contain WordNet antonyms. To evaluate the precision of our method, we randomly selected 200 examples of rules eliminated from DIRT (irrespective of the textual entailment data) and a human evaluator decided if they are indeed incorrect inference rules. Out of these 92% turned out to be incorrect rules, such as *X right about Y* $\approx$ *X wrong about Y*. However, there are also cases of correct rules being eliminated, such as *X* ***have*** *second thoughts about Y* $\approx$ *X* ***lack*** *confidence about Y*.

# 5 Inference Rules on Tree Skeletons

In order to address the issues described in Section 3.1, we choose to apply the rule collection on a dependency-based representation of **T** and **H**. We will first introduce this representation and the algorithm to derive it, and following that we will describe how we applied the inference rules on this structure.

**Tree Skeletons**

The Tree Skeleton (TS) structure was proposed by [Wang and Neumann, 2007], and can be viewed as an extended version of the predicate-argument structure. Since it contains not only the predicate and its arguments, but also the dependency paths in-between, it captures the essential part of the sentence.

Following their algorithm, we first preprocess the data using the Minipar dependency parser and then select overlapping topic words (i.e. nouns) in **T** and **H**

(we use fuzzy match at the substring level instead of full match). Starting with these nouns, we traverse the dependency tree to identify the lowest common ancestor node (named as *root node*). This sub-tree without the inner yield is defined as a Tree Skeleton. Figure 1 shows the TS of **T** in the pair:

**T** *For their discovery of ulcer-causing bacteria, Australian doctors Robin Warren and Barry Marshall have received the 2005 Nobel Prize in Physiology or Medicine.*

**H** *Robin Warren was awarded a Nobel Prize.*

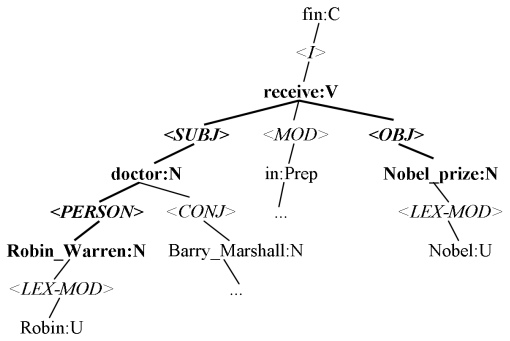


Figure 1: Dependency structure of text. Tree skeleton in bold

Notice that, in order to match the inference rules with two anchors, the number of the dependency paths from the nouns to the root node should also be two. In practice, tree skeletons can be extracted from approximately 30% of the T-H pairs.

**Applying DIRT on a TS**

After extracting the TS, the next step is to find the inference rules which match the two tree skeletons of a T-H pair. This is done in a rather straightforward manner. Given tree skeletons of **T** and **H**, we check if the two left dependency paths, the two right ones or the two root nodes contain the patterns of a rule.

In the example above, the rule $X \xleftarrow{obj} receive \xrightarrow{subj} Y \approx X \xleftarrow{obj2} award \xrightarrow{obj1} Y$ satisfies this criterion, as it is matched at the root nodes. Notice that the rule is correct only in restricted contexts, in which the object of *receive* is something which is conferred on the basis of merit.

# 6 Experiments

Our experiments consist in predicting positive entailment in a very straightforward rule-based manner. For each collection we select the RTE pairs in which we find

| RTE Set | $\text{Dirt}_{TS}$ | Dirt+$\text{WN}_{TS}$ | $\text{Id}_{TS}$ | Dirt+Id+$\text{WN}_{TS}$ | Dirt+Id+$\text{WN}^*_{TS}$ |
|---|---|---|---|---|---|
| RTE2 | 55/0.63 | 103/0.65 | 45/0.66 | 136/0.65 | 90/0.67 |
| RTE3 | 48/0.66 | 79/0.65 | 29/0.79 | 101/0.69 | 74/0.71 |

Table 3: Results on tree skeletons with various rule collections

a tree skeleton and match an inference rule. The first number in our table entries represents how many of such pairs we have identified, out of 1600 development and test pairs. For these pairs we simply predict positive entailment and the second entry represents what percentage of these pairs are indeed true entailment. Our work does not focus on building a complete RTE system but we also combine our method with a bag of words baseline to see the effects on the entire data set.

In the first two columns (Table 3: $\text{Dirt}_{TS}$ and Dirt+$\text{WN}_{TS}$) we consider DIRT in its original state and DIRT with rules generated with WordNet as described in Section 4; all precisions are higher than 63%[4]. After adding WordNet, tree skeletons and rules are matched in approximately twice as many pairs, while the precision is not harmed. This may indicate that our method of adding rules does not decrease precision of an RTE system.

In the third column we report the results of using a set of rules containing only the trivial identity ones ($\text{Id}_{TS}$). For our current system, this can be seen as a $precision$ upper bound for all the other collections, in concordance with the fact that identical rules are nothing but inference rules of highest possible confidence. The fourth column (Dirt+Id+$\text{WN}_{TS}$) contains what can be considered our best setting. In this setting three times as many pairs are covered using a collection containing DIRT and identity rules with WordNet extension. Although the precision results with this setting are encouraging (65% for RTE2 data and 69% for RTE3 data), the coverage is still low, 8% for RTE2 and 6% for RTE3. This aspect together with an error analysis we performed are the focus of Section 7.

Another experiment aimed at improving the precision of our predictions. For this we further restrict our method: we have a true entailment only if applying the inference rule to a TS leaves no unmatched lexical items in the fragment of the dependency path where it has been identified. The more restricted method (Dirt+Id+$\text{WN}^*_{TS}$) gives, as expected, better precision with an approximately 30% loss in coverage.

At last, we also integrate our method with a bag of words baseline, which calculates the ratio of overlapping words in **T** and **H**. For the pairs that our method covers, we overrule the baseline's decision. The results are shown in Table 4. On

[4]The RTE task is considered to be difficult. The average accuracy of the systems in the RTE-3 challenge is around 61% [Giampiccolo et al., 2007]

the full data set, the improvement is still small due to the low coverage of our method, however on the pairs that are covered by our method, there is a significant improvement over the overlap baseline.

| RTE Test(# pairs) | BoW | BoW&Main |
|---|---|---|
| RTE2 (89) | 52.80% | 60.67% |
| RTE2 (800) | 56.87% | 57.75% |
| RTE3 (71) | 52.11% | 59.15% |
| RTE3 (800) | 61.12% | 61.75% |

Table 4: Results on RTE test data. Covered set and full set.

| Source of error | # pairs | % pairs |
|---|---|---|
| TS structure | 7 | 23% |
| Incorrect rules | 9 | 30% |
| Other | 14 | 47% |

Table 5: Error analysis

## 7 Discussion

In this section we take a closer look at the data in order to better understand how does our method of combining tree skeletons and inference rules work.

For error analysis we consider the pairs incorrectly classified in the RTE3 data, consisting of a total of 30 pairs. We classify the errors into three main categories: tree skeleton structure errors, inference rule errors, and other errors (Table 5).

In the first category, seven T-H pairs are incorrect. In those cases the tree skeleton fails to match the corresponding anchors of the inference rules. For instance, if someone founded *the Institute of Mathematics (Instituto di Matematica) at the University of Milan*, it does not follow that they founded *The University of Milan*.

Approximately 30% of the errors are caused by incorrect inference rules. Out of these, two are correct in some contexts but not in the entailment pairs in which they are found. For example, the following rule *X generate Y* $\approx$ *X earn Y* is used incorrectly, however in the restricted context of $money$ or $income$, the two verbs have similar meaning. An example of an incorrect rule is *X issue Y* $\approx$ *X hit Y* since it is difficult to find a context in which this holds.

The last category contains all the other errors. In all these cases, the additional information conveyed by the text or the hypothesis which cannot be captured by our current approach, affects the entailment. For example *an imitation diamond* is not a $diamond$, and *more than 1,000 members of the Russian and foreign media* does not entail *more than 1,000 members from Russia*; these are not trivial, since lexical semantics and fine-grained analysis of the restrictors are needed.

In a second part of our analysis we discuss the coverage issue, based on an analysis of uncovered pairs. A main factor in failing to detect pairs in which entailment rules should be applied is the fact that the tree skeleton does not find the

corresponding lexical items of two rule patterns. In one of the pairs *78% increase in X* entails *X rose by 78%*. This rule is available, however the tree skeletons capture $reach$ and $rise$ as key verbal nodes. In another example, information such as the fact that $rains$ are *creating flooding* and *devastating* are all necessary to conclude that *floods are ravaging Europe*. However a structure like tree skeleton cannot capture all these elements. Issues will occur even if the tree skeleton structure is modified to align all the corresponding fragments together. Consider constructions with embedding verbs such as *manage*, *forget*, *attempt*. Our method can detect if the two embedded verbs convey a similar meaning, however not how the embedding verbs affect the entailment. Independent of the shortcomings of our tree skeleton structure, a second factor in failing to detect true entailment still lies in lack of rules (e.g. the last two examples in Table 1 are entailment pair fragments which can be formulated as inference rules, but are not straightforward to acquire).

# 8 Conclusion

Throughout the paper we have identified important issues encountered in using inference rules for recognizing textual entailment and proposed methods to solve them. We explored the possibility of combining a collection obtained in a statistical, unsupervised manner, DIRT, with a hand-crafted lexical resource in order to make inference rules have a larger contribution to applications. We also investigated ways of effectively applying these rules. The experiment results show that although coverage is still not satisfying, the precision is promising. Therefore our method has the potential to be successfully integrated into a larger entailment detection framework.

The error analysis points out several possible future directions. The tree skeleton representation we used needs to be enhanced in order to capture more accurately the relevant fragments of the text. A different issue remains the fact that a lot of rules we could use for RTE are still lacking. A proper study of the limitations of the DH as well as a classification of the knowledge we want to encode as inference rules would be a step forward towards solving this problem. Furthermore, although all the inference rules we used aim at recognizing positive entailment cases, it is natural to use them for detecting negative cases of entailment as well. In general, we can identify pairs in which the patterns of an inference rule are present but the anchors are missmatched, or they are not in the correct hypernym/hyponym relation. This can be the base of a principled method for detecting structural contradictions [de Marneffe et al., 2008].

# References


[Bar-Haim et al., 2007] Bar-Haim, R., Dagan, I., Greental, I., Szpektor, I., and Friedman, M. (2007). Semantic inference at the lexical-syntactic level for textual entailment recognition. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 131–136, Prague.

[Barzilay and McKeown, 2001] Barzilay, R. and McKeown, K. R. (2001). Extracting paraphrases from a parallel corpus. In *Proceedings of 39th Annual Meeting of the Association for Computational Linguistics*, pages 50–57, Toulouse, France.

[Clark et al., 2007] Clark, P., Harrison, P., Thompson, J., Murray, W., Hobbs, J., and Fellbaum, C. (2007). On the role of lexical and world knowledge in rte3. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 54–59.

[Dagan et al., 2006] Dagan, I., Glickman, O., and Magnini, B. (2006). The pascal recognising textual entailment challenge. In *Lecture Notes in Computer Science, Vol. 3944, Springer*, pages 177–190. Quionero-Candela, J.; Dagan, I.; Magnini, B.; d'Alch-Buc, F. Machine Learning Challenges.

[de Marneffe et al., 2008] de Marneffe, M.-C., Rafferty, A. N., and Manning, C. D. (2008). Finding contradictions in text. In *Proceedings of ACL-08: HLT*, pages 1039–1047, Columbus, Ohio.

[Giampiccolo et al., 2007] Giampiccolo, D., Magnini, B., Dagan, I., and Dolan, B. (2007). The third pascal recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 1–9, Prague.

[Harabagiu and Hickl, 2006] Harabagiu, S. and Hickl, A. (2006). Methods for using textual entailment in open-domain question answering. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL*, pages 905–912, Sydney, Australia.

[Iftene and Balahur-Dobrescu, 2007] Iftene, A. and Balahur-Dobrescu, A. (2007). Hypothesis transformation and semantic variability rules used in recognizing textual entailment. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 125–130, Prague.

[Lin, 1998] Lin, D. (1998). Dependency-based evaluation of minipar. In *Proc. Workshop on the Evaluation of Parsing Systems*, Granada.

[Lin and Pantel, 2001] Lin, D. and Pantel, P. (2001). Dirt. discovery of inference rules from text. In *KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 323–328, New York, USA.

[Marsi et al., 2007] Marsi, E., Krahmer, E., and Bosma, W. (2007). Dependency-based paraphrasing for recognizing textual entailment. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 83–88, Prague.

[Pang et al., 2003] Pang, B., Knight, K., and Marcu, D. (2003). Syntax-based alignment of multiple translations: Extracting paraphrases and generating new sentences. In *HLT-NAACL*, pages 102–109.

[Szpektor et al., 2007] Szpektor, I., Shnarch, E., and Dagan, I. (2007). Instance-based evaluation of entailment rule acquisition. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 456–463, Prague, Czech Republic.

[Szpektor et al., 2004] Szpektor, I., Tanev, H., Dagan, I., and Coppola, B. (2004). Scaling web-based acquisition of entailment relations. In *Proceedings of EMNLP*, pages 41–48.

[Wang and Neumann, 2007] Wang, R. and Neumann, G. (2007). Recognizing textual entailment using sentence similarity based on dependency tree skeletons. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 36–41.