Finding contradictions in text

Marie-Catherine de Marneffe

1 Introduction

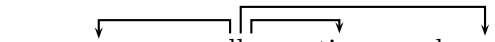
In this paper, I seek to understand the ways contradictions occur across texts and I describe a system for automatically detecting such constructions. Finding conflicting statements is foundational for text understanding, a problem which recently received a surge of interest in the computational linguistics community. Condoravdi *et al.* (2003) first recognized the importance of handling both entailment and contradiction for text understanding: "relations of entailment and contradiction are the key data of semantics, as traditionally viewed as a branch of linguistics. The ability to recognize such semantic relations is clearly not a *sufficient* criterion for language understanding: there is more than just being able to tell that one sentence follows from another. But we would argue that it is a minimal, *necessary* criterion." (p. 38).

So far however, work in robust text understanding has focused on entailment: systems aimed at providing textual inference in arbitrary domains.[1] The task of textual inference first appeared latent within the field of question answering (Pasca & Harabagiu 2001; Moldovan *et al.* 2003). As schematized in Figure 1, the question *What company sells most greetings cards?* can be viewed as a statement containing a variable (*what company*) which in this case is of the ORGANIZATION type. If the system finds a text passage that entails the statement and contains a possible assignment for the variable (mainly a concept of the same type), the variable assignment is taken as the answer to the question. In this example, the passage *Hallmark remains the largest maker of greetings cards* entails the question, *Hallmark* is of the type ORGANIZATION, and will be the answer to the question. The task of textual inference then received attention within the PASCAL Recognizing Textual Entailment (RTE) Challenges (Dagan *et al.* 2006; Bar-Haim *et al.* 2006; Giampiccolo *et al.* 2007), and related work within the U.S. Government AQUAINT program. In the RTE challenges, systems are given pairs of sentences, called *text* (T) and *hypothesis* (H), and the goal is to identify whether the hypothesis follows from the text and general background knowledge, according to the intuitions of an intelligent human reader. That is, the standard is not whether the hypothesis is logically entailed, but whether it can reasonably be inferred: "We say that T entails H if the meaning of H can be inferred from the meaning of T, as would typically be interpreted by people. This somewhat informal definition is based on (and assumes) common human understanding of language as well as common background knowledge." (Dagan *et al.* 2005, p. 1).

[1] Work in knowledge-based systems has looked extensively at knowledge-base consistency (Preece 1994).

QUESTION:

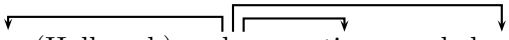
What company sells most greetings cards?



ORGANIZATION sells greetings cards most

ANSWER:

Hallmark remains the largest maker of greetings cards



ORGANIZATION(Hallmark) maker greetings cards largest

Figure 1: Example of Question/Answer treatment (from Pasca & Harabagiu 2001).

Two main sets of approaches have been explored for text understanding. One set of methods builds on work in formal semantics, translating sentences into first-order logic (FOL), and applying then a theorem prover or a model builder (Fowler *et al.* 2005; Bos & Markert 2006). Such approaches focus on precise semantic interpretation and attain high precision, but do so at the cost of very poor recall. These methods based on formal semantics provide deep but brittle text understanding. They often collapse because of the difficulty of translating natural language into FOL, and do not scale well to arbitrary text. However, for textual understanding to be useful, we want systems which are open-domain and can degrade gracefully in the presence of incomplete or inaccurate semantic representations. Hence, the other type of approaches aims at shallow but robust text understanding. These approaches use impoverished semantic representations: some rely on measures of lexical and semantic overlap (Jijkoun & de Rijke 2005), while others operate on semantic graphs derived from syntactic dependency parses (Hickl *et al.* 2006; MacCartney *et al.* 2006). But even without deep understanding, they have proven useful in determining entailment. The work presented in this paper follows the trend of providing a robust system, at the cost of semantic precision.

Besides being a necessary step on the way to text understanding, contradiction detection also has a number of applications in information analysis and updating. Automatic detection of contradictions has the potential to highlight areas of contention and differences among positions. Consider applying a contradiction detection system to political candidate debates. By drawing attention to topics in which candidates have conflicting positions, such a system could enable voters to make more informed choices between candidates and sift through the ever growing amount of available information. Contradiction detection could also be applied to intelligence reports, demonstrating which information is least certain.

The goal of this paper is to shed light on the complex picture of contradiction in text. First, I provide a definition of contradiction that is suitable for natural language processing (NLP) applications, as well as a collection of contradiction corpora. Analyzing these data, I find contradiction is a rare phenomenon that may be created in a number of ways; I propose

a typology of contradiction classes and tabulate their frequencies. Contradictions may arise from relatively obvious features such as antonymy, negation, or numeric mismatches. But, they also arise from complex differences in the structure of assertions, discrepancies based on world knowledge, and lexical contrasts. Consider the following pair of sentences:

(1) Police specializing in explosives defused the rockets. Some 100 people were working inside the plant.

(2) 100 people were injured.

This pair forms a contradiction due to a series of cause and effect relations: if rockets are defused, they cannot go off and thus cannot injure anyone. Contrast this to detecting entailments. Here, it is relatively easy to identify the lack of entailment: the first sentence involves no injuries, so the second is unlikely to be entailed. Detecting contradictions thus appears to be a harder task than detecting entailments. Most entailment systems operate relatively successfully using a weak proof theory (Hickl *et al.* 2006; MacCartney *et al.* 2006; Zanzotto *et al.* 2007), but contradictions require deeper inferences and model building: more precise comprehension of the consequences of sentences is crucial. Moreover, detecting contradictions requires event coreference: for texts to contradict, they must refer to the same event. The importance of event coreference was recognized in the MUC information extraction tasks which targeted identifying scenarios related to the same event (Humphreys *et al.* 1997). While recent work in text understanding has not focused on this issue, which does not appear for textual entailment detection (since unrelated sentences are not entailed), it must be tackled in a successful contradiction system.

The system described here includes event coreference, and I present the first detailed examination of contradiction detection performance on corpora that include all types of contradictions in the proposed typology.

2 Related work

Little work has been conducted on contradiction detection. The recent PASCAL RTE Challenges focused on inference, but in the context of the last competition, a pilot experiment was conducted using the RTE3 data, in which systems made a 3-way decision as to whether pairs of sentences were entailed, contradictory, or neither.[2]

As already mentioned, Condoravdi *et al.* (2003) first emphasized that contradiction, as well as entailment, needs to be considered to provide robust text understanding. However, they restrict these phenomena to their logical definition (see the discussion in Section 3.1). They use a contexted clausal representation derived from approaches in formal semantics, but do not report any empirical results for their system.

To my knowledge, Harabagiu *et al.* (2006) give the first empirical results for contradiction detection, but focus on specific kinds of contradiction: those featuring overt negation as well as those formed by paraphrases. They constructed two corpora on which they evaluated their system. One (LCC_negation) was created by overtly negating each entailment in the RTE2

[2]Information about this task as well as data can be found at http://nlp.stanford.edu/RTE3-pilot/.

data, producing a balanced dataset. To avoid overtraining, negative markers were also added to each instance of non-entailment while ensuring that these markers did not create contradictions. Their second corpus (LCC_paraphrase) was produced by paraphrasing the hypothesis sentences from LCC_negation, removing the negation: *A hunger strike was not attempted* was loosely paraphrased by *A hunger strike was called off*. They achieved very good performance: accuracies of 75.63% on LCC_negation and 62.55% on LCC_paraphrase. Yet, contradictions are not limited to these constructions; to be practically useful, any system must aim to provide broader coverage.

3 Contradictions

In this section, I propose an appropriate definition of contradiction for NLP applications, and a typology of contradictions which emerges from data analysis of the corpora I developed.

3.1 What is a contradiction?

One standard for defining the term 'contradiction' is to consider sentences A and B contradictory if there is no possible world in which A and B are both true; that is, a strict logical condition of contradiction. While this definition is easy to apply, it misses many pairs of sentences that humans would find contradictory. For contradiction detection to be useful, it is necessary to match the intuitions of an intelligent human reader, identifying sentences that are extremely unlikely to both be true at the same time. Thus, pairs of sentences such as *Sally sold a boat to John* and *John sold a boat to Sally* are tagged as contradictory even though it could be that each sold a boat to the other. This looser definition captures human intuitions of "incompatiblity," and perfectly fits applications that seek to highlight discrepancies in descriptions of the same event. Examples of contradiction are given in table 1.

For texts to be contradictory, they must involve the same event. Two phenomena need be considered to make this determination: implied coreference and embedded texts. Given limited context, whether two entities are coreferent may be probable rather than certain. Because I attempt to match the intuitions of a human reader concerning likely contradiction, compatible noun phrases between sentences are assumed to be coreferent in the absence of clear countervailing evidence. In the following example, the *woman* in the first and second sentences is not necessarily the same, but one would likely assume it is if the two sentences appeared together, creating a contradiction:

(3) Passions surrounding Germany's final match at the Euro 2004 soccer championships turned violent when a woman stabbed her partner in the head because she didn't want to watch the game on television.

(4) A woman passionately wanted to watch the soccer championship.

I also mark as contradictions pairs of texts reporting contradictory statements. The following sentences can be viewed as referring to the same event (*de Menezes in a subway station*), and display incompatible views of this event:

ID	Type	Text	Hypothesis
1	Antonym	Capital punishment is a catalyst for more crime.	Capital punishment is a deterrent to crime.
2	Negation	A closely divided Supreme Court said that juries and not judges must impose a death sentence.	The Supreme Court decided that only judges can impose the death sentence.
3	Numeric	The tragedy of the explosion in Qana that killed more than 50 civilians has presented Israel with a dilemma.	An investigation into the Israeli strike in Qana found 28 confirmed dead thus far.
4	Factive	Turkey is unlikely to become involved in, or allow U.S. forces to use Turkish territory in a Middle East war that does not threaten her territory directly.	U.S. to use Turkish military bases.
5	Factive	The bombers had not managed to enter the embassy compounds.	The bombers entered the embassy compounds.
6	Structure	Jacques Santer succeeded Jacques Delors as president of the European Commission in 1995.	Delors succeeded Santer in the presidency of the European Commission.
7	Structure	The Channel Tunnel stretches from England to France. It is the second-longest rail tunnel in the world, the longest being a tunnel in Japan.	The Channel Tunnel connects France and Japan.
8	Linguistic	The Canadian parliament's Ethics Commission said former immigration minister, Judy Sgro, did nothing wrong and her staff had put her into a conflict of interest.	The Canadian parliament's Ethics Commission accuses Judy Sgro.
9	Linguistic	In the election, Bush called for U.S. troops to be withdrawn from the peacekeeping mission in the Balkans.	He cites such missions as an example of how America must "stay the course."
10	WK	Microsoft Israel was founded in 1989 and became one of the first Microsoft branches outside the USA.	Microsoft was established in 1989.

Table 1: Examples of contradiction types.

(5) That police statement reinforced published reports, that eyewitnesses said de Menezes had jumped over the turnstile at Stockwell subway station and was wearing a padded jacket.

(6) However, the documents leaked to ITV News suggest that Menezes walked casually into the subway station and was wearing a light denim jacket.

We can see these examples as carrying an "embedded contradiction." Contrary to Zaenen *et al.* (2005), I argue that recognizing embedded contradictions, regardless of the source, is important for the application of a contradiction detection system: if *John thinks that he is incompetent*, and *his boss believes that John is not being given a chance*, one would like to detect that the targeted information in the two sentences is contradictory, even though logically the two sentences can both be true at the same time.

3.2 Guidelines for contradiction annotation

I developed guidelines for annotating contradiction which were used for my own data annotation and were also used by assessors at NIST for annotating the RTE3 pilot experiment. The guidelines, as distributed, are given below.

Recognizing Textual Entailment (RTE) items consist of two pieces of text, a brief text and a short hypothesis. For some, the hypothesis follows from the text (that is, a normal reader would be happy to accept the text as strong evidence that the hypothesis is true, assuming that the text is reliable). This is technically referred to as "entailment". These items are marked "YES". You shouldn't change these. For the rest, we wish to distinguish between whether the text and hypothesis are contradictory, which we will label "NO", or whether the two pieces contain overlapping or different information but the hypothesis neither follows from or contradicts the text, which we will label "UNKNOWN".

Definition of contradiction

To decide if the text and hypothesis are contradictory, ask yourself the following question: If I were shown two contemporaneous documents one containing each of these passages, would I regard it as very unlikely that both passages could be true at the same time? If so, the two contradict each other. Another way of stating this would be: the hypothesis is contradictory if assertions in the hypothesis appear to directly refute, or show portions of the text to be false/wrong, if the hypothesis were taken as reliable. You should be able to state a clear basis for a contradiction, such as "the text says the group traveled west to Mosul, while the hypothesis says they were traveling from Syria (which is to the east of Mosul)."

For example, the following are contradictions:

[RTE-1 828] contradiction

T: Jennifer Hawkins is the 21-year-old beauty queen from Australia.

H: Jennifer Hawkins is Australia's 20-year-old beauty queen.

[RTE-2 404] contradiction

T: In that aircraft accident, four people were killed: the pilot, who was wearing civilian clothes, and three other people who were wearing military uniforms.

H: Four people were assassinated by the pilot.

You should mark as a contradiction a text and hypothesis reporting contradictory statements, if the reports are stated as facts. We can see these as carrying an embedded contradiction. For example:
[RTE-2 320] contradiction

T: That police statement reinforced published reports, that eyewitnesses said de Menezes had jumped over the turnstile at Stockwell subway station and was wearing a padded jacket, despite warm weather.

H: However, the documents leaked to ITV News suggest that Menezes, an electrician, walked casually into the subway station and was wearing a light denim jacket.

For something to be a contradiction, it does not have to be impossible for the two reports to be reconcilable, it just has to appear highly unlikely in the absence of further evidence. For instance, it is reasonable to regard the first pair below as a contradiction (it is not very plausible that the bodies (of someone who has a secretary, etc.) were not found for over 18 months), but it does not seem prudent to regard the second pair as contradictory (despite a certain similarity in the reports, they could easily both be true):
[RTE-1 579] contradiction

T: The anti-terrorist court found two men guilty of murdering Shapour Bakhtiar and his secretary Sorush Katibeh, who were found with their throats cut in August 1991.

H: Shapour Bakhtiar died in 1989.

[RTE-1 2113] unknown: not a contradiction

T: Five people were killed in another suicide bomb blast at a police station in the northern city of Mosul.

H: Five people were killed and 20 others wounded in a car bomb explosion outside an Iraqi police station south of Baghdad.

How to interpret the data?
(1) Noun phrase coreference:
Compatible noun phrases between the text and the hypothesis should be treated as coreferent in the absence of clear countervailing evidence. For example, below we should assume that the two references to "a woman" refer to the same woman:
[RTE-1 201] contradiction

T: Passions surrounding Germany's final match at the Euro 2004 soccer championships turned violent when a woman stabbed her partner in the head because she didn't want to watch the game on television.

H: A woman passionately wanted to watch the soccer championship.

Similarly, references to dates like "Thursday" should be assumed to be coreferent in the absence of countervailing evidence.

(2) Event coreference:
Whether to regard a text and hypothesis as describing the same event is more subtle. If two descriptions appear overlapping, rather than completely unrelated, by default assume that the two passages describe the same context, and contradiction is evaluated on this basis. For example, if there are details that seem to make it clear that the same event is being described, but one passage says it happened in 1985 and the other 1987, or one passage says two people met in Greece, and the other in Italy, then you should regard the two as a contradiction. Below, it seems reasonable to regard "a ferry collision" and "a ferry sinking" as the same event, and then the reports make contradictory claims on casualties:
[RTE-2 237] contradiction

T: Rescuers searched rough seas off the capital yesterday for survivors of a ferry collision that claimed at least 28 lives, as officials blamed crew incompetence for the accident.

H: 100 or more people lost their lives in a ferry sinking.

In other circumstances, it is most reasonable to regard the two passages as describing different events. You have to make your best judgment, given the limited information available. You should use world knowledge about the frequency of event types in making this decision. For instance, example RTE-1 2113 above was not marked as a contradiction, as it does not seem compelling to regard "another suicide bomb blast" and "a car bomb explosion" as referring to the same event. And for the two passages below, there just doesn't seem much evidence that they have anything to do with each other:
[RTE-2 333] unknown (not a contradiction)

T: The European-born groups with the highest labor force participation rates were from Bosnia and Herzegovina.

H: The European country with the highest birth rate is Bosnia-Herzegovina.

In the general RTE guidelines, it says the text and the hypothesis are meant to be regarded as roughly contemporaneous, but may differ in date by a few days, and so details of tense are meant to be ignored when deciding whether a text entails the hypothesis or not. However in an example like the following, it seems clear that the hypothesis is not possible as a consistent, contemporaneous statement with the text, and so we mark it as contradictory.
[RTE-3 357] contradiction

T: The Italian parliament may approve a draft law allowing descendants of the exiled royal family to return home. The family was banished after the Second World War because of the King's collusion with the fascist regime, but moves were introduced this year to allow their return.

H: Italian royal family returns home.

Contradictions in RTE data

For past RTE data sets, contradictions represent about 30% of the non-entailment RTE data. This isn't a target for the data you will annotate, but is just meant to give you some idea of what to expect.

We have made available RTE3_dev data annotated for the 3-way classification of YES, UNKNOWN, and NO. In this data, the texts and decisions of YES are unchanged from the (revised) RTE3_dev data (no matter if occasional errors still lurk therein). The answers that were previously "NO" were subclassified as to whether they were contradictions (still "NO") or not – that is, unrelated or incomplete informational overlap (now "UNKNOWN").

3.3 Typology of contradictions

Contradictions may arise from a number of different constructions, some overt and others that may be complex even for humans to detect. Analyzing contradiction corpora (see section 3.4), I find that there are primarily two categories of contradiction: (1) those occurring via antonymy, negation, and numeric mismatch (date or number mismatches), which are relatively simple to detect, and (2) contradictions arising from the use of factive or modal words (Factive/Modal), structural (Structure) and subtle lexical contrasts (Linguistic), as well as world knowledge (WK). Table 1 gives examples of the different contradiction types.

I consider contradictions in category (1) to be "easy" because they can often be automatically detected without full sentence comprehension. For example, if words in the two passages are antonyms and the sentences are reasonably similar, especially in polarity, a contradiction occurs. Additionally, little external information is needed to gain broad coverage of antonymy, negation, and numeric mismatch contradictions; each involves only a closed set of words or data that can be obtained using existing resources and techniques (e.g., WordNet (Fellbaum 1998), VerbOcean (Chklovski & Pantel 2004)).

However, contradictions in category (2) are more difficult to detect automatically because they require complex and precise models of sentence meaning. For instance, to find the contradiction in example 8 (table 1), it is necessary to learn that *X said Y did nothing wrong* and *X accuses Y* are incompatible. Presently, there exist methods for learning oppositional terms (Marcu & Echihabi 2002) and paraphrase learning has been thoroughly studied, but successfully extending these techniques to learn incompatible phrases poses difficulties because of the data distribution. Example 9 provides an even more difficult instance of contradiction created by a lexical discrepancy. Structural issues also create contradictions, as in examples 6 and 7. Lexical complexities and variations in the function of arguments across verbs can make recognizing these contradictions complicated. Even when similar verbs are used and clear argument differences exist, structural differences may indicate either non-entailment or contradiction, and distinguishing the two automatically is problematic. Consider contradiction 7 in table 1 and the following pair, which is not a contradiction:

Data	# contradictions	# total pairs
RTE1_dev1	48	287
RTE1_dev2	55	280
RTE1_test	149	800
RTE2_dev	111	800
RTE3_dev	80	800
RTE3_test	72	800

Table 2: Number of contradictions in the RTE datasets.

(7) The CFAP purchases food stamps from the federal government and distributes them to eligible recipients.

(8) A government purchases food.

In both cases, the first sentence discusses one entity (*CFAP*, *The Channel Tunnel*) which has a relationship (*purchase*, *stretch*) to other entities. The second sentence posits a similar relationship that includes one of the entities involved in the original relationship as well as an entity that was not involved. However, different outcomes result because a tunnel can only connect two unique locations whereas more than one entity may purchase food. These frequent interactions between world knowledge and structure make it hard to ensure that any particular instance of structural mismatch is a contradiction.

3.4 Contradiction corpora

Following the guidelines given in section 3.2, I annotated the existing RTE datasets for contradiction. These datasets contain pairs consisting of a short text followed by a one-sentence hypothesis. Table 2 gives the number of contradiction pairs found in each dataset. The RTE datasets are roughly balanced between entailments and non-entailments, and even in these constructed datasets targeting inference, the number of contradictions is low, as most instances of non-entailments are unrelated or partially overlapping texts. The RTE3_test dataset was independently annotated by NIST as part of the RTE3 pilot task in which systems made a 3-way decision as to whether pairs of sentences were entailed, contradictory, or neither.

My annotations and those of NIST were performed on the original RTE datasets, contrary to Harabagiu *et al.* (2006). Because their corpora are constructed using negation and paraphrase, they are unlikely to cover all types of contradictions detailed in section 3.3. We might hypothesize that rewriting explicit negations commonly occurs via the substitution of antonyms. Imagine for instance:

H:	Bill has finished his math.
Negated H:	Bill hasn't finished his math.
Paraphrase of Negated H:	Bill is still working on his math.

The rewriting in both the negated and the paraphrased corpora is likely to leave one in the space of "easy" contradictions and addresses fewer than 25% of contradictions (table 3). I contacted

	Type	RTE3_dev	'Real' corpus
1	Antonym	15.0	7.5
	Negation	8.8	10.0
	Numeric	8.8	33.8
2	Factive/Modal	5.0	5.0
	Structure	16.3	3.8
	Linguistic	18.8	25.0
	WK	27.5	15.0

Table 3: Percentages of contradiction types in RTE3_dev and the real contradiction corpus.

the LCC authors to obtain their datasets, but they were unable to make them available to me. Thus, I simulated the LCC_negation corpus, adding negative markers to the RTE2_test data (Neg_test), and to a development set (Neg_dev) constructed by randomly sampling 50 pairs of entailments from the RTE2 development dataset, as well as 50 pairs of non-entailments.

In the annotation of the RTE datasets, I did not explicitly construct contradictions, but this corpus still does not reflect 'real-life' contradictions. I therefore also collected contradictions 'in the wild.' The resulting corpus contains 80 pairs of contradictory passages: 19 from newswire, mainly looking at related articles in Google News, 51 from Wikipedia, in which the article editing history sometimes indicates the reason for the change, and 10 from the Lexis Nexis database. Despite the randomness of the collection, I argue that this corpus best reflects contradictions that naturally occur in texts.

Table 3 gives the percentages of each type of contradiction for the RTE3_dev dataset and the real contradiction corpus. Globally, we see that contradictions in category (2) occur frequently, and even dominate the RTE dataset. In the real contradiction corpus, there is a much higher rate of the numeric and linguistic types of contradiction. This supports the intuition that in the real world, contradictions primarily occur for two reasons: information is updated as more knowledge of an event is acquired over time (e.g., a rising death toll) or various parties have divergent views of an event based on their own perspectives (see example 9 in table 1).

All the contradiction corpora—the simulation of the LLC_negation corpus, the RTE datasets and the real contradictions—will be made publicly available.

4 System overview

The contradiction detection system presented in this paper uses a stage-architecture similar to the Stanford RTE system (MacCartney *et al.* 2006; Chambers *et al.* 2007) and other various RTE systems (Hickl *et al.* 2006; Rodrigo *et al.* 2007), but adds a stage for event coreference decision. The common stages in RTE systems are: linguistic preprocessing, alignment of words between the text and the hypothesis, and finally extraction of entailment features. In the contradiction detection system, the extra stage for event coreference decision is added before feature extraction.

4.1 Stage 1: Linguistic analysis

The first stage computes linguistic representations of the text and the hypothesis that contain as much information as possible about their semantic content. The text and hypothesis are converted into typed dependency graphs produced by the Stanford parser (Klein & Manning 2003; de Marneffe *et al.* 2006). To improve the dependency graph as a pseudo-semantic representation, collocations in WordNet and named entities are collapsed. This way, entities and multiword relations become a single node in the graph. The nodes in the final dependency graph are annotated with their associated word, part-of-speech (given by the parser), lemma and named-entity tag (given by a CRF-based NER tagger).

4.2 Stage 2: Alignment between graphs

The second stage provides an alignment between the hypothesis and text graphs, consisting of a mapping from each node in the hypothesis graph to a unique node in the text graph or to null. Figure 2 gives an example of graph alignment between the following text/hypothesis pair:

T: CNN reported that several troops were killed in today's ambush.

H: Thirteen soldiers lost their lives in the ambush.

The scoring measure is designed to favor alignments which align semantically similar subgraphs, irrespective of polarity. Therefore nodes receive high alignment scores when the words they represent are semantically similar. The scoring metric takes into account the word, the lemma, and the part-of-speech, and searches for word relatedness using external resources, such as WordNet, precomputed latent semantic analysis matrices, and special-purpose gazettes. Alignment scores also incorporate structural information based on the shape of the paths between nodes in the text graph which correspond to adjacent nodes in the hypothesis graph. Similarity measures and structural information are combined via weights learned using the passive-aggressive online learning algorithm MIRA (Crammer & Singer 2001).

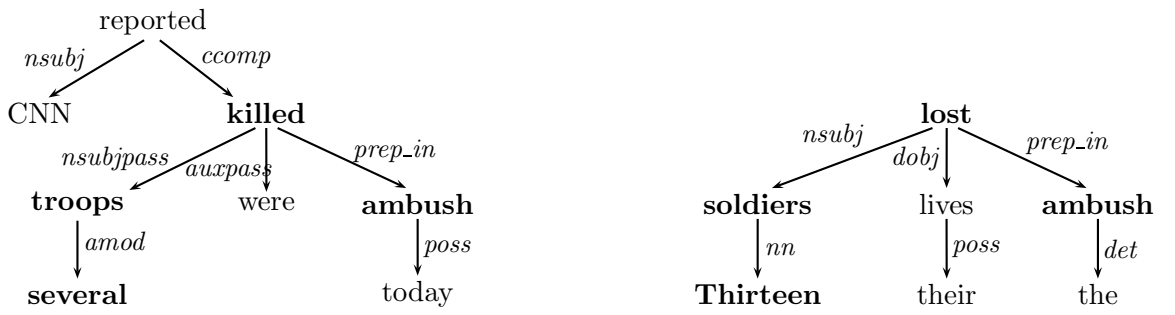
4.3 Stage 3: Filtering non-coreferent events

The contradiction features that are extracted in the last stage of the system look for mismatches between the text and hypothesis. Therefore, an important step is to first remove pairs of sentences which do not describe the same event, and thus cannot be contradictory to one another. In the following example, it is necessary to recognize that *new moon* is not the same entity as *the moon Titan*; otherwise the conflicting diameters result in the system labeling the pair as contradictory.

T: The new moon, which is only about 25 miles in diameter, was photographed 13 years ago.

H: The moon Titan has a diameter of 5100 kms.

This issue does not arise for textual entailment: elements in the hypothesis not supported by the text lead to non-entailment, regardless of whether the same event is described. For contradiction,



Alignment:			
	lost	→	**killed**
	soldiers	→	**troops**
	Thirteen	→	**several**
	ambush	→	**ambush**

Figure 2: Dependency graphs of text T *CCN reported that several troops were killed in today's ambush* and hypothesis H *Thirteen soldiers lost their lives in the ambush*, as well as alignment from hypothesis to text.

however, it is critical to filter unrelated sentences to avoid finding false evidence of contradiction when there is contrasting information about different events.

Given the structure of RTE data, in which the hypotheses are shorter and simpler than the texts, one straightforward strategy for detecting coreferent events is to check whether the root of the hypothesis graph is aligned in the text graph. However, some RTE hypotheses are testing systems' abilities to detect relations between entities, such as in the following examples:

T: The automobile industry, led by General Motors and Ford, offered price discounts from June-August that temporarily boosted sales, but these discounts may have masked underlying weakness.

H: Ford is part of the automobile industry.

T: Almost immediately upon his return from the war in December 1944, George Bush married Barbara Pierce.

H: The name of George H.W. Bush's wife is Barbara.

Thus, I do not filter verb roots that are indicative of such relations. As shown in table 4, this strategy of filtering (Root) improves results on RTE data, compared to no filter at all. For real world data, however, the assumption of directionality made in this strategy is unfounded. Moreover, we cannot assume that one sentence will be short and the other more complex. Assuming two sentences of comparable complexity, I hypothesize that modeling topicality could be used to assess whether the sentences describe the same event.

By the strictest definition, there is a continuum of topicality from the start to the end of a sentence (Firbas 1971). I thus originally defined the topicality of a NP by n^w where n is the

Strategy	Precision	Recall
No filter	55.10	32.93
Root	61.36	32.93
Root + topic	61.90	31.71

Table 4: Precision and recall for contradiction detection on RTE3_dev using different filtering strategies.

*n*th NP in the sentence. Additionally, I accounted for multiple clauses by weighting each clause equally; in example 4 in table 1, *Australia* receives the same weight as *Prime Minister* because each begins a clause. However, this weighting was not supported empirically, and I thus use a simpler, unweighted model. The topicality score of a sentence is calculated as a normalized score across all aligned NPs.[3] The text and hypothesis are topically related if either sentence score is above a tuned threshold. Modeling topicality provides an additional improvement in precision (table 4).

While filtering provides improvements in performance, some examples of non-coreferent events are still not filtered, such as the two following pairs:

T: Also Friday, five Iraqi soldiers were killed and nine wounded in a bombing, targeting their convoy near Beiji, 150 miles north of Baghdad.

H: Three Iraqi soldiers also died Saturday when their convoy was attacked by gunmen near Adhaim.

T: Five people were killed in another suicide bomb blast at a police station in the northern city of Mosul.

H: Five people were killed and 20 others wounded in a car bomb explosion outside an Iraqi police station south of Baghdad.

It seems that the real world frequency of events needs to be taken into account. In this case, attacks in Iraq are unfortunately frequent enough to assert that it is unlikely that the two sentences present mismatching information (i.e., different location) about the same event. But compare the following example:

T: President Kennedy was assassinated in Texas.

H: Kennedy's murder occurred in Washington.

In this case, the two sentences refer to one same unique event, and the location mismatch renders them contradictory.

[3]Since dates can often be viewed as scene setting rather than what the sentence is about, I ignore these in the model. However, ignoring or including dates in the model creates no significant differences in performance on RTE data.

4.4 Stage 4: Extraction of contradiction features

In the final stage of the system, contradiction features are extracted, on which I apply a logistic regression to classify the pair as contradictory or not. The feature weights are hand-set, guided by linguistic intuition. These features are described in detail in the next section.

5 Features for contradiction detection

In this section, I define each of the feature sets used in the system, which attempt to capture salient patterns of contradiction.

Number, date and time features. A numeric mismatch can indicate contradiction as in example 3 (table 1), or as in the following pair where the date mismatch creates a contradiction:

T: In 1955, the Aurora Borealis, a coating applied to crystal stones to produce a "rainbow of colors" effect was introduced.

H: The Aurora Borealis was discovered in 1993.

The numeric features are therefore designed to recognize (mis-)matches between numbers, dates, and times. Date and time expressions are normalized, and numbers are represented as ranges. This also includes expression matching (e.g., *over 100* and *200* will not be considered a mismatch). Aligned numbers are marked as mismatches only when they are incompatible and the words governing them match well. Ensuring that the numbers refer to the same entity is crucial when looking for contradiction, but is not necessary when detecting non-entailment. Indeed mismatching information will often be cue for non-entailment, as in the following example where the dates do not match (*1968* vs. *1969*):

T: On March 10, 1969, James Earl Ray was sentenced to 99 years in a Tennessee prison after he pleaded guilty to the murder of Martin Luther King Jr., but for blacks that hardly seemed recompense.

H: Martin Luther King was murdered in 1968.

However the example above is not a contradiction. Since the system realizes that the dates are not related to the same verb (*murdered* vs. *sentenced*), it will not take this numeric mismatch as evidence for contradiction.

Structural features. These features aim to determine whether the syntactic structures of the text and hypothesis create contradictory statements. For example, the subjects and objects for each aligned verb are compared. If the subject in the text overlaps with the object in the hypothesis, this is evidence for a contradiction. Consider example 6 in table 1. In the text, the subject of *succeed* is *Jacques Santer* while in the hypothesis, *Santer* is the object of *succeed*, suggesting that the two sentences are incompatible. Another pattern targeted is contradictions arising from a superlative modified by an ordinal:

T: The slender tower is the second tallest building in Japan.

H: The slender tower is the tallest building in Japan.

As discussed in section 3.3, many structural mismatches may indicate either contradiction or non-entailment depending on the context, so the contradiction detection system is able to target only a limited number of structural features with high precision.

Polarity features. Polarity difference between the text and hypothesis is often a good indicator of contradiction, provided there is a good alignment (see example 2 in table 1). The polarity features capture the presence (or absence) of linguistic markers of negative polarity contexts. These markers are scoped such that words are considered negated if they have a negation dependency in the graph or are an explicit linguistic marker of negation (e.g., simple negation (*not*), downward-monotone quantifiers (*no*, *few*), or restricting prepositions). In some cases, negations are also propagated onto the governor, such as in the following hypothesis:

T: Heemeyer cut portholes for his guns, then welded a tight enclosure around them so police bullets could not penetrate.

H: Out of an old bulldozer, Heemeyer built a concrete box that no police bullet could penetrate.

The negation *no* in the constituent *no police bullet* is propagated onto the verb *penetrate*. This allows to correctly assess that the verbs *penetrate* in the text and the hypothesis have the same polarity.

Polarity difference is estimated for a word in the hypothesis and its aligned word in the text: if one word is negated and the other is not, there might be a polarity difference. This difference is confirmed by checking that the two words are not antonyms and that they lack unaligned prepositions or other context suggesting that they do not refer to the same thing.

Antonymy features. Aligned antonyms are a very good cue for contradiction. The list of antonyms and contrasting words comes from WordNet, from which I extract words with direct antonymy links and expand the list by following synonymy links. I also use oppositional verbs from VerbOcean. I check whether an aligned pair of words appear in the list, as well as checking for common antonym prefixes (e.g., *anti*, *un*). The polarity of the context is used to determine if the presence of antonyms creates a contradiction.

Modality features. These capture simple patterns of modal reasoning. The text and the hypothesis are mapped to one of six modalities, according to the presence of predefined modality markers such as *can* or *maybe*: (*not_*)*possible*, (*not_*)*actual*, (*not_*)*necessary*. A contradiction feature is produced if the text/hypothesis modality pair gives rise to a contradiction. For instance, the following pair will be mapped to the contradiction judgment (*possible*, *not_possible*):

T: The trial court may allow the prevailing party reasonable attorney fees as part of costs.

H: The prevailing party may not recover attorney fees.

A priori the modality pair (*possible*, *actual*) should give rise to contradiction, as in:

T: Sonia Gandhi can be defeated in the next elections in India by BJP.

H: Sonia Gandhi is defeated by BJP.

However as pointed out by Manning (2006), *may* or *can* are also used as a form of hedging, especially in scientific or political discourse.

T: Suncreams designed for children could offer less protection than they claim on the bottle.

H: Suncreams designed for children protect at the level they advertise.

The system cannot handle this distinction of use, and I therefore chose not to classify the modality pair (*possible*, *actual*) as a marker of contradiction.

Factivity features. The context in which a verb phrase is embedded may give rise to contradiction, as in example 5 (table 1). Negation influences some factivity patterns for contradiction: *Bill forgot to take his wallet* contradicts *Bill took his wallet* while *Bill did not forget to take his wallet* does not contradict *Bill took his wallet*. To capture this, I expanded the PARC lists of factive, implicative and non-factive verbs (Nairn *et al.* 2006), and cluster them according to how they create contradiction. I then determine to which class the (grand)parent of the text aligned with the hypothesis root belongs to, and generate features accordingly.

Relational features. A large proportion of the RTE data is derived from information extraction tasks where the hypothesis captures a relation between elements in the text. Using a pattern matching language for dependency graphs called Semgrex, I am able to find such relations and ensure that the arguments between the text and the hypothesis match. In the following example, the system would see that *Fernandez* works for *FEMA*, and that because of the negation, a contradiction arises.

T: Fernandez, of FEMA, was on scene when Martin arrived at a FEMA base camp.

H: Fernandez doesn't work for FEMA.

Relational features provide accurate information but are difficult to extend to create broad coverage.

6 Results

The contradiction detection system was developed on all datasets listed in the first part of table 5. As test sets, I used RTE1_test, the independently annotated RTE3_test, and Neg_test. I focused on attaining high precision. In a real world setting, it is likely that the contradiction rate is extremely low; rather than overwhelming true positives with false positives, rendering the system impractical, contradictions are marked conservatively.

	Precision	Recall	Accuracy
RTE1_dev1	70.37	40.43	–
RTE1_dev2	72.41	38.18	–
RTE2_dev	64.00	28.83	–
RTE3_dev	61.90	31.71	–
Neg_dev	74.07	78.43	75.49
Neg_test	62.97	62.50	62.74
LCC_negation	–	–	75.63
RTE1_test	42.22	26.21	–
RTE3_test	22.95	19.44	–
Avg. RTE3_test	10.72	11.69	–

Table 5: Precision and recall figures for contradiction detection on all corpora. Accuracy is given for balanced datasets only. 'LCC_negation' refers to performance of Harabagiu *et al.* 2006; 'Avg. RTE3_test' refers to mean performance of the 12 submissions to the RTE3 pilot.

	Type	RTE3_dev		RTE3_test	
1	Antonym	25.0	(3/12)	42.9	(3/7)
	Negation	71.4	(5/7)	60.0	(3/5)
	Numeric	71.4	(5/7)	28.6	(2/7)
2	Factive/Modal	25.0	(1/4)	10.0	(1/10)
	Structure	46.2	(6/13)	21.1	(4/19)
	Linguistic	13.3	(2/15)	0.0	(0/12)
	WK	18.2	(4/22)	8.3	(1/12)

Table 6: Recall by contradiction type.

The results on the test sets show that performance drops on new data, highlighting the difficulty in generalizing from a small corpus of positive contradiction examples, as well as underlining the complexity of building a broad coverage system. This drop in accuracy on the test sets is greater than that of many RTE systems, suggesting that generalizing for contradiction is more difficult than for entailment. Particularly when addressing contradictions that require linguistic and world knowledge, I am able to only add coverage in a piecemeal fashion, resulting in improved performance on the development sets but in small gains for the test sets. Thus, as shown in table 6, the system achieves 13.3% recall on linguistic contradictions in RTE3_dev but is unable to identify any such contradictions in RTE3_test. Table 7 gives the percentages of contradiction types found in RTE3_test, which roughly pattern the percentages in RTE3_dev.

Additionally, I found that the precision of category (2) features was less than that of category (1) features. Structural features, for example, made the system tag 36 non-contradictions as contradictions in RTE3_test, over 75% of the precision errors.

As RTE3_test has been independently annotated by NIST, I also annotated that corpus, and

	Type	RTE3_dev	RTE3_test
1	Antonym	15.0	9.7
	Negation	8.8	6.9
	Numeric	8.8	9.7
2	Factive/Modal	5.0	13.9
	Structure	16.3	26.4
	Linguistic	18.8	16.7
	WK	27.5	16.7

Table 7: Percentages of contradiction type in RTE3_dev and RTE3_test.

found a high inter-annotator agreement ($\kappa = 0.81$), showing that, even when limited context is available, humans tend to agree on what a contradiction is.

7 Error analysis and discussion

One significant issue in contradiction detection is the lack of feature generalization. This problem is especially apparent for items in category (2) requiring linguistic and world knowledge, which proved to be the most difficult contradictions to detect on a broad scale. While the system is able to find certain specific relationships in the development sets, these features attained only limited coverage. Many contradictions in this category require multiple inferences and remain beyond the system's capabilities, as illustrated by the following pairs, as well as example 10 in table 1:

T: The Auburn High School Athletic Hall of Fame recently introduced its Class of 2005 which includes 10 members.

H: The Auburn High School Athletic Hall of Fame has ten members.

T: In 1833 Van Hasselt left Maastricht, then blockaded by the Belgian forces, and made his way to Brussels, where he became a naturalized Belgian, and was attached to the Bibliotheque de Bourgogne.

H: Van Hasselt was of Belgian origin.

Of the types of contradictions in category (2), the system is best at addressing those formed via structural differences and factive/modal constructions as shown in table 6. For instance, the system correctly identifies examples 5 and 6 in table 1 as contradictions. Another example is the factive contradiction the system correctly identifies in RTE3_test:

T: Marcel Beaubien unsuccessfully sought election to the Canadian House of Commons as the Conservative candidate in the federal riding of Sarnia-Lambton in 2004.

H: Marcel Beaubien was elected to the Canadian House of Commons.

However, creating features with sufficient precision is an issue for these types of contradictions. Intuitively, two sentences that have aligned verbs with the same subject and different objects (or vice versa) are contradictory. In the following par, for example, the aligned verbs *received* have the same object (*title of "Newcastle-under-Lyne"*), but different subjects (*John Holles* vs. *Pelham*):

T: The title was again created for John Holles. When he died in 1711 the title became extinct but his estates passed to his nephew Thomas Pelham, who three years later upon coming of age received the title in its third creation. In 1757 Pelham received the additional title of "Newcastle-under-Lyne".

H: John Holles received the title of "Newcastle-under-Lyne".

This kind of structural mismatch indicates a contradiction 55% of the time on the development sets, but this is not high enough precision given the rarity of contradictions. For example, the following pair is erroneously marked as contradictory:

T: A serial killer on Mumbai's streets killed yet another person last night; making the latter one of five young men murdered recently.

H: Five men have been killed by a serial killer in Mumbai.

The system identifies a structural mismatch because the aligned verbs *killed* have the same logical subject (*serial killer*) but different objects (*another person* vs. *five men*).

Another type of contradiction where precision falters is numeric mismatch. The system obtains high recall for this type (table 6), as it is relatively simple to determine if two numbers are compatible and refer to the same entity, but high precision is difficult to achieve due to differences in what numbers may mean. Consider:

T: Nike Inc. said that its profit grew 32 percent, as the company posted broad gains in sales and orders.

H: Nike said orders for footwear totaled $4.9 billion, including a 12 percent increase in U.S. orders.

The system detects a mismatch between *32 percent* and *12 percent*, ignoring the fact that one refers to *profit* and the other to *orders*. Accounting for context requires extensive comprehension of the text; here, it is not sufficient to simply look at whether the two numbers are headed by similar words (*grew* and *increase*). The system also lacks calculation, which is sometimes necessary to realize that numbers are in fact not incompatible:

T: In Rwanda there were on average 8,000 victims per day for about 100 days.

H: There were 800,000 victims of the massacres in Rwanda.

Another issue is the quality of the alignment between the hypothesis and text graphs. In some cases, a bad alignment, and not the contradiction detection mechanism per se, is the cause of recall or precision errors:

T: Since Concorde's first flight in 1969, it was recognized as the safest airplane in the history of aviation. And in spite of this dramatic crash on July 25, it still remains the safest way to fly.

H: Concorde's first crash was in 1969.

In this pair, *crash* in the hypothesis is aligned to *flight* in the text, which are identified as oppositional terms, and the system therefore incorrectly marks the pair as contradictory.

T: Prime Minister John Howard says he will not be swayed by a videotaped warning that Australia faces more terrorism attacks unless it withdraws its troops from Iraq and Afghanistan.

H: Australia withdraws from Iraq.

In this example, the system fails to find the contradiction. For unclear reasons, *withdraws* in the hypothesis is not aligned to *withdraws* in the text. If it was, the system would realize that there is a polarity mismatch between the text (*withdraws* is under the scope of *unless*) and the hypothesis (where *withdraws* is in a neutral context). Finding the optimal alignment between the hypothesis and the text is a difficult search process, and this paper does not focus on this issue. However, improvement in the alignment stage would clearly increase the performance of the system.

As demonstrated by the 63% accuracy it achieves on Neg_test, the system is reasonably good at detecting negation and correctly ascertaining whether it is a symptom of contradiction. The instances on which the system fails are often complex, such as:

T: Even today, within the deepest recesses of our mind, lies a primordial fear that will not allow us to enter the sea without thinking about the possibility of being attacked by a shark.

H: A shark attacked a human being.

Here, the system incorrectly thinks that *attacked by a shark* is negated because of the presence of *without thinking*, and since the hypothesis is not negated, a contradiction is detected. However, globally, contradictions arising from negation are well handled by the system. Similarly, single word antonymy is tackled with high precision (78.9%). Harabagiu *et al.*'s performance demonstrates that further improvement on these types is possible; indeed, they use more sophisticated techniques to extract oppositional terms and detect polarity differences. Thus, detecting category (1) contradictions is feasible with current systems.

While these category (1) contradictions form only a third of those in the RTE datasets, detecting such contradictions accurately would solve half of the problems found in the real corpus. This suggests that sufficient traction on contradiction detection for real world applications may be gained. Even so, category (2) contradictions must be targeted to detect many of the most interesting examples and to solve the entire problem of contradiction detection. One obvious way to improve recall of the world knowledge type is to incorporate some geographic knowledge into the system. There are available resources of such information, and contradictions such as the following might then be detected:

T: Pythagoras was born in Ionia on the island of Samos, and eventually settled in Crotone, a Dorian Greek colony in southern Italy, in 529 B.C.E. There he lectured in philosophy and mathematics.

H: Pythagoras was born in Crotone.

Another way to find more contradictions would be to expand the list of oppositional terms, using techniques similar to Harabagiu *et al.* (2006). For example, the current list of antonyms does not contain the pair *bougth/gifted*, and therefore the system cannot find the following contradiction:

T: The Crathes castle served as the ancestral seat of the Burnetts of Leys until gifted to the National Trust for Scotland by the 13th Baronet of Leys, Sir James Burnett in 1951.

H: The Crathes caste was bought by the National Trust for Scotland.

Not only oppositional terms need to be learned, but also "exclusion terms":

T: Most of the life of Petko Kiryakov was not unveiled by historians but by the prominent Bulgarian writer Nikolai Haitov, who wrote a novel and a script which was turned into a TV series, which became a favourite of most Bulgarians.

H: Nikolai Haitov is a historian.

In this case, the contradiction arises because Nikolai Haitov is presented as a writer, not as an historian, and usually people have one profession. Despite the problem of data distribution, it would be worthwhile investigating how expanding existing methods for learning oppositional terms would fare in learning exclusion terms.

8 Conclusion

This paper aimed at investigating the concept of contradiction for text understanding. I proposed a suitable definition of contradictions for NLP applications, which captures human intuitions of what a contradiction is. The data annotated using this definition constitutes the biggest contradiction corpus publicly available so far.

I constructed a typology of contradictions emerging from the data. The different types can be grouped in two categories, according on how easily they can be automatically detected: category (1) contains contradictions which are relatively easy to find, arising from negation, antonymy or numeric mismatch; category (2) groups contradictions that necessitate much deeper comprehension, arising from structural mismatches, lexical contrasts or discrepancies based on world knowledge.

This paper underlines how finding contradiction and detecting entailment, both foundational tasks for text understanding, differ. I showed that a system for contradiction needs to make more fine-grained and subtle distinctions than the more common systems for entailment. In particular, I argued that assessing event coreference is a crucial step for contradiction detection, and I incorporated such a component into the system.

I presented the first detailed breakdown of performance on this task. Despite the fact that some types of contradiction, such as linguistic and world knowledge, are currently beyond the system's grasp, it achieves good performance on other types such as those arising from negation and antonymy. Overall the investigation presented here demonstrates which aspects of contradiction detection can be resolved and where further research must be directed.

Acknowledgments

Most of these results have been submitted to ACL 2008 as joint work with Anna Rafferty and Christopher D. Manning. I wish to thank them both for their help and contribution to the ideas and system presented here.

References

Bar-Haim, Roy, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, & Idan Szpektor. 2006. The second PASCAL recognising textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, Venice, Italy.

Bos, Johan, & Katja Markert. 2006. When logical inference helps determining textual entailment (and when it doesn't). In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*.

Chambers, Nathanael, Daniel Cer, Trond Grenager, David Hall, Chloe Kiddon, Bill MacCartney, Marie-Catherine de Marneffe, Daniel Ramage, Eric Yeh, & Christopher D. Manning. 2007. Learning alignments and leveraging natural logic. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*.

Chklovski, Timothy, & Patrick Pantel. 2004. Verbocean: Mining the web for fine-grained semantic verb relations. In *Proceedings of EMNLP-04*.

Condoravdi, Cleo, Dick Crouch, Valeria de Pavia, Reinhard Stolle, & Daniel G. Bobrow. 2003. Entailment, intensionality and text understanding. *Workshop on Text Meaning (2003 May 31)* .

Crammer, Koby, & Yoram Singer. 2001. Ultraconservative online algorithms for multiclass problems. In *Proceedings of COLT-2001*.

Dagan, Ido, Oren Glickman, & Bernardo Magnini. 2006. The PASCAL recognising textual entailment challenge. In *MLCW 2005, LNAI Volume 3944*, ed. by Quinonero-Candela et al., 177–190. Springer-Verlag.

de Marneffe, Marie-Catherine, Bill MacCartney, & Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC-06*.

Fellbaum, Christiane. 1998. *WordNet: an electronic lexical database*. MIT Press.

Firbas, Jan. 1971. On the concept of communicative dynamism in the theory of functional sentence perspective. *Brno Studies in English* 7.23–47.

Fowler, Abraham, Bob Hauser, Daniel Hodges, Ian Niles, Adrian Novischi, & Jens Stephan. 2005. Applying COGEX to recognize textual entailment. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment*.

Giampiccolo, Danilo, Ido Dagan, Bernardo Magnini, & Bill Dolan. 2007. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*.

Harabagiu, Sanda, Andrew Hickl, & Finley Lacatusu. 2006. Negation, contrast, and contradiction in text processing. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence (AAAI-06)*.

Hickl, Andrew, John Williams, Jeremy Bensley, Kirk Roberts, Bryan Rink, & Ying Shi. 2006. Recognizing textual entailment with LCC's GROUNDHOG system. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*.

Humphreys, Kevin, Robert Gaizauskas, & Saliha Azzam. 1997. Event coreference for information extraction. In *Proceedings of the Workshop on Operational Factors in Pratical, Robust Anaphora Resolution for Unrestricted Texts, 35th ACL meeting*.

Jijkoun, Valentin, & Maarten de Rijke. 2005. Recognizing textual entailment using lexical similarity. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment*.

Klein, Dan, & Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Meeting of the Association of Computational Linguistics*.

MacCartney, Bill, Trond Grenager, Marie-Catherine de Marneffe, Daniel Cer, & Christopher D. Manning. 2006. Learning to recognize features of valid textual entailments. In *Proceedings of the North American Association of Computational Linguistics (NAACL-06)*.

Manning, Christopher D., 2006. Local textual inference: it's hard to circumscribe, but you know it when you see it – and NLP needs it. ms.

Marcu, Daniel, & Abdessamad Echihabi. 2002. An unsupervised approach to recognizing discourse relations. In *Proceedings of the 4Oth Meeting of the Association for Computational Linguistics*.

Moldovan, Dan, Christine Clark, Sanda Harabagiu, & Steven Maiorano. 2003. COGEX: A logic prover for question answering. In *Proceedings of the North American Association of Computational Linguistics (NAACL-03)*.

Nairn, Rowan, Cleo Condoravdi, & Lauri Karttunen. 2006. Computing relative polarity for textual inference. In *Proceedings of ICoS-5*.

Pasca, Marius, & Sanda Harabagiu. 2001. High performance question/answering. In *Proceedings of SIGIR 01*.

Preece, Alun. 1994. Validation of knowledge-based systems: The state-of-the-art in North America. *The Journal for the Integrated Study of Artificial Intelligence Cognitive Science and Applied Epistomology* 11(4).

Rodrigo, Alvaro, Anselmo Peñas, Jesus Herrera, & Felisa Verdejo. 2007. Experiments of UNED at the third recognising textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*.

Zaenen, Annie, Lauri Karttunen, & Richard S. Crouch. 2005. Local textual inference: can it be defined or circumscribed? In *ACL 2005 Workshop on Empirical Modeling of Semantic Equivalence and Entailment*.

Zanzotto, Fabio Massimo, Marco Pennacchiotti, & Alessandro Moschitti. 2007. Shallow semantics in fast textual entailment rule learners. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*.