Building an Annotated Textual Inference Corpus for Motion and Space

Kirk Roberts
Human Language Technology Research Institute
University of Texas at Dallas
Richardson TX 75080
kirk@hlt.utdallas.edu

Abstract

This paper presents an approach for building a corpus for the domain of motion and spatial inference using a specific class of verbs. The approach creates a distribution of inference features that maximize the discriminatory power of a system trained on the corpus. The paper addresses the issue of using an existing textual inference system for generating the examples. This enables the corpus annotation method to assert whether more data is necessary.

1 Introduction

Open-domain textual inference provides a vast array of challenges to a textual entailment system. In order to ensure a wide distribution of these challenges in building the PASCAL 2005 corpus (Dagan et al., 2005), seven different application settings were used for inspiration: Information Retrieval, Comparable Documents, Reading Comprehension, Question Answering, Information Extraction, Machine Translation, and Paraphrase Acquisition. While PASCAL 2005 and its subsequent challenges have released numerous corpora for open-domain textual inference, many types of textual inference are sparsely represented. This speaks not to a weakness in the mentioned corpora, but rather the depth and complexity of challenges that textual inference presents.

Furthermore, the open-domain inference task often forces systems to face more than one of these challenges on a single inference pair (such as requiring both an understanding of paraphrases and part-whole relationships). In many cases, it is desirable to isolate out most of these "subtasks" within textual inference and concentrate on a single aspect. Partly for this reason, the Boeing-Princeton-ISI (BPI) Textual Entailment Test Suite[1] was developed. Its focus is real-world knowledge and not syntactic constructions, so it provides 250 syntactically simple but semantically rich inference pairs.

This paper explores the creation of such a specific textual inference corpus based on verb classes, specifically focusing on the class of *motion verbs* and their nominalizations. The goal is to develop a publicly available corpus for spatial inference involving *motion*. Section 2 analyzes the properties of such a corpus. Section 3 outlines the effort to build a motion corpus. Finally, Section 4 discusses considerations for the size of the corpus.

2 Properties of an Inference Corpus

2.1 General Properties

Annotated corpora are designed for training and evaluation for specific classification tasks, and thus an optimal corpus is one that maximizes a system's ability to form a discriminative feature space. However, knowing ahead of time what the feature space will look like may be difficult. But, at the same time the corpus should be also reflective of the real world.

One method for developing a useful corpus under these conditions, especially for a specific domain, is to use an existed textual entailment system that can aid in the example generation process. By using such a system to suggest examples, one is able to both reduce the time (and cost) of annotation as well as producing a corpus with a desirable distribution of features.

[1] Available at http://www.cs.utexas.edu/∼pclark/bpi-test-suite/

Proceedings of the 2009 Workshop on Applied Textual Inference, ACL-IJCNLP 2009, pages 48–51,
Suntec, Singapore, 6 August 2009. ©2009 ACL and AFNLP

Text: John flew to New York from LA. **Hypothesis**: John left LA for New York.
Text: John will fly over the Atlantic during his trip to London from New York on Tuesday. **Hypothesis**: On Tuesday, John flew over water when going from North America to Europe.

Table 1: Examples of textual inference for motion.

2.2 Properties of a Motion Corpus

Textual inference about motion requires an external representation apart from the text. While many inference pairs can be solved with strategies such as lexical alignment or paraphrasing, many texts assume the reader has knowledge of the properties of motion. Table 1 shows two such inference pairs. The first can be solved through a paraphrase strategy, while the second requires explicit knowledge of the properties of motion that are difficult to acquire through a paraphrasing method. Unfortunately, most open-domain inference corpora are sparsely populated with such types of inference pairs, so a new corpus is required.

For the purpose of the corpus, the concept of motion is strictly limited to the set of words in the (Levin, 1993) verb-class MOTION. This greatly benefits the annotation process: passages or sentences without a verb or nominalization that fits into the MOTION class can immediately be discarded. Levin's verb classes are easily accessible via VERBNET (Kipper et al., 1998), which provides additional syntactic and semantic information as well as mappings into WORDNET (Fellbaum, 1998).

(Muller, 1998) proposes a qualitative theory of motion based on spatio-temporal primitives, while (Pustejovsky and Moszkowicz, 2008) shows an annotation structure for motion. Furthermore, representing motion requires the complete representation of spatial information, as motion is simply a continuous function that transforms space. (Hobbs and Narayanan, 2002) discuss many of the properties for spatial representation, including dimensionality, frame of reference, regions, relative location, orientation, shape, and motion. It is therefore desirable for a motion corpus to require inference over many different aspects of space as well as motion. Table 2 shows the properties of motion incorporated in the inference system.

In practice, these properties are far from uniformly distributed. Properties such as *dest*(M_x) are far more common than *shape*(M_x). Clearly, having a system that performs well on destinations is more important than one that can draw inferences from motion's effects on an object's shape ("*the car hit the barricade and was crushed*"), but it is still desirable to have a corpus that provides systems with examples of such properties.

Property	Description
motion(M_x)	Instance of motion in text
theme(M_x)	Object under motion
area(M_x)	Area of motion
src(M_x)	Source location
dest(M_x)	Destination location
path(M_x)	Path of motion
current(M_x)	Current position
orientation(M_x)	Direction/Orientation
shape(M_x)	Shape of object
t_start(M_x)	Start of motion
t_end(M_x)	End of motion

Table 2: Extracted properties of motion.

The corpus annotation process shall disregard many discourse-related phenomena, including coreference. Further, the text and hypothesis for each inference pair will be limited to one sentence. In this way, knowledge of motion is emphasized over other linguistic tasks.

3 Building a Corpus Focusing on Knowledge about Motion

To build the motion inference corpus, we chose to start with an existing, large document corpus, AQUAINT-2.[2] This corpus is composed of 2.4GB of raw files and contains over 900,000 documents. Having a large corpus is important for finding sparse verbs like *escort* and *swing* and sparse properties like *area*(M_x) and *orientation*(M_x).

3.1 Text Annotation

In order to get a more diverse distribution of motion verbs and properties (hereafter, just referred to as properties) than the given distribution from the corpus, the following procedure is considered:

Let V_s be the (static) distribution of motion properties from the document corpus. Let V_d be the (dynamic) distribution of motion properties from an (initially empty) set of annotated examples. Next, define a "feedback" distribution V_f, such that for each property y:

$$P_f(y) = \frac{max(0, 2P_s(y) - P_d(y))}{Z} \tag{1}$$

Where $P_s(y)$, $P_d(y)$, and $P_f(y)$ are the probabilities of property y in distributions V_s, V_d, and

[2] Available through the Linguistic Data Consortium, id LDC2008T25

V_f, respectively, and Z is a normalization factor (needed when the numerator is zero).

Let the parameter α determine the likelihood of sampling from this distribution V_f or from the uniform distribution U. The function *NextExampleType(V_f, α)* then specifies which motion property should be in the next example. An unannotated example is then drawn from an index, annotated by the user, and placed in the set of annotated examples. V_d is then updated to reflect the new distribution of verbs and properties in the annotated example set.

There are several items to note. First, the example might contain multiple properties not chosen by the *NextExampleType* method. When a motion event with a *path*(M_x) is chosen, it is not uncommon for a *dest*(M_x) property to be a part of the same event. This is why the V_d and V_f distributions are necessary: they are a feedback mechanism to try to keep the actual distribution, V_d, as close to the desired distribution as possible.

Second, the value for α is the sole pre-specified parameter. It dictates the likelihood of choosing an example despite its *a priori* probability. Setting α to 1.0 will result in only sampling based on the V_f distribution, and setting it to 0.0 will generate a uniform sampling. In practice, this is set to 0.8 to allow many of the sparse features through.

Third, V_d and V_f account even for properties generated from the uniform distribution. In practice this means that low-probability events will be generated from U and not V_f, especially later in the sampling process. Due to the non-independence of the properties as discussed above, this discrepancy is difficult to account for and is considered acceptable: U will still dictate a much higher distribution of low-probability properties than would otherwise be the case.

3.2 Hypothesis Annotation

While the hypothesis itself must be written by the annotator, one can apply some of the same principles to ensure a coverage of motion concepts. Since not every motion term in the text need be tested by the hypothesis, it is beneficial to keep track of which properties are tested within each. For this reason, the annotator is responsible for indicating which motion properties are used in the hypothesis. This way, the annotator can be alerted to any properties under-represented in the set of hypotheses relative to the set of annotated texts.

Feature	#	Seq	Ex Gen
dest(M_x)	749	48	60
go	382	90	129
leave	105	376	454
...	...	...	...
orientation(M_x)	94	420	282
flee	4	9,991	5,508
steer	2	20,000	7,065
parachute	1	40,000	8,227

Table 3: Motion features with instance counts from 2000 sample sentences. The *Seq* (Sequential) and *Ex Gen* (see Section 3.1) columns are the expected number of annotated sentences for 20 instances of the feature to be found using that method, assuming i.i.d.

3.3 Evaluation

The purpose of the algorithm from Section 3.1 is not only to build a more balanced corpus, but to do so more quickly. By looking through examples that are more likely to maintain a balanced corpus, annotators are saved from looking through hundreds (or thousands!) of examples that contain overly redundant properties.

To illustrate this point, consider a random sample of 2000 sentences. Table 3 shows the extracted counts for some of the least and most common verbs and properties alongside projections of how many motion sentences would need to be annotated with and without the algorithm to attain a rather modest 20 examples of each. The results prove that, for many features, the example generation approach allows many more instances of that feature to be placed in the corpus.

3.4 Comparison with Active Learning

The process presented in Section 3.1 bears a close resemblence with active learning, so the differences between the two merit some discussion. Active learning seeks to improve a specific classifier by selecting training data based on some confidence/score metric for the purpose of improving an overall metric (usually the score across all annotated data). Often, examples on which the classifier is the least confident are presented to an annotator for manual classification. Then the system is re-trained to include the new data, and the process repeats.

The annotation process presented above, however, is not "active" in this same sense. Instead it seeks a certain distribution of properties regardless of a classifier's ability to accurately perform inferences. The primary advantage, then, is a corpus that is not designed for a specific classification

Corpus	# Dev	# Test
RTE-1	567	800
RTE-2	800	800
RTE-3	800	800
BPI	250	

Table 4: Number of annotated inferences for each inference corpus.

technique or set of features. A secondary advantage is that it avoids the risk of choosing poor examples but rather seeks a breadth of data.

4 Corpus Size Considerations

An important consideration–and an active area of research–is the ideal size of an annotated corpus. As one can see from Table 4, the RTE tasks make 800 examples available for an open-domain textual inference corpus.

But when the scope of the corpus is more limited, perhaps 800 examples is too few or too many. If the intent is to provide a set on which systems can be blindly evaluated for motion inference, then a much smaller number is required than a corpus intended for training machine-learned models. In this case, we seek to do the latter.

It should be mentioned that if the corpus generation process follows the algorithm presented in Section 3.1, then any reasonable number of inference pairs should follow the same distribution as a much larger set. For this reason, it is possible to adopt the active learning approach and build the corpus incrementally by iteratively annotating until satisfactory results are reached or gains are minimal.

5 Discussion

In addition to building a motion-specific corpus, this paper argues for the creation of domain-specific corpora for textual inference. Beyond simply measuring a system's ability to reason for specific tasks, they enable the aquisition of world knowledge through training data. They can then be used by statistical learning techniques applied to natural language processing. This is different than generating axioms and using them in abductive reasoning, which is another approach to approximate world knowledge.

Levin's verb classes (of which there are less than fifty) are a useful way to organize corpora. Levin's classes are structured under the assumption that syntactic and semantic frames are directly linked within each class. Since all verbs within the class have similar semantic arguments, knowledge aquisition becomes manageable. A system that has a wide coverage of knowledge trained on such corpora could claim a wide coverage of knowledge of all verb-based events within text.

6 Conclusion

This paper has presented an argument for the creation of domain-specific textual inference corpora and, in general terms, what that corpus should look like. In particular, it has described the ongoing process of building an inference corpus for spatial inference about motion. It has shown how an existing system can be used to aid in the example generation and annotation process with analysis as to the effects of the algorithm on presenting more balanced data. Finally, the paper discussed some considerations for the size of such a corpus.

Upon completion, the corpus will be made publically available.

References

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognizing textual entailment challenge. In *Proceedings of the First PASCAL Challenges Workshop on Recognising Textual Entailment*, pages 1–8.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Jerry R. Hobbs and Srini Narayanan. 2002. Spatial representation and reasoning. In *Intelligent Systems: Concepts and Applications*, pages 67–76. MacMillan.

Karin Kipper, Hoa Trang Dang, and Martha Palmer. 1998. Class-based construction of a verb lexicon. In *In Proceedings of AAAI/IAAI*.

Beth Levin. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. The University of Chicago Press.

Philippe Muller. 1998. A Qualitative Theory of Motion Based on Spatio-Temporal Primitives. In *KR '98: Principles of Knowledge Representation and Reasoning*, pages 131–141.

James Pustejovsky and Jessica L. Moszkowicz. 2008. Integrating Motion Predicate Classes with Spatial and Temporal Annotations. In *Proceedings of COLING 2008*, pages 95–98.