Use of Modality and Negation in Semantically-Informed Syntactic MT

Kathryn Baker[*]
U.S. Department of Defense

Michael Bloodgood[**]
University of Maryland

Bonnie J. Dorr[†]
University of Maryland

Chris Callison-Burch[‡]
Johns Hopkins University

Nathaniel W. Filardo[‡]
Johns Hopkins University

Christine Piatko[§]
Johns Hopkins University

Lori Levin[||]
Carnegie Mellon University

Scott Miller[#]
BBN Technologies

[*] U.S. Department of Defense, 9800 Savage Rd., Suite 6811, Fort Meade, MD 20755. E-mail: kathrynlb@gmail.com.

[**] Center for Advanced Study of Language, University of Maryland, 7005 52nd Avenue, College Park, MD 20742. E-mail: meb@umd.edu.

[†] Department of Computer Science and UMIACS, University of Maryland, AV Williams Building 3153, College Park, MD 20742. E-mail: bonnie@umiacs.umd.edu.

[‡] Center for Language and Speech Processing, Johns Hopkins University, 3400 N. Charles Street, Hackerman Hall 320, Baltimore MD 21218. E-mail: {ccb,nwf}@cs.jhu.edu.

[§] Applied Physics Laboratory, Johns Hopkins University, 11000 Johns Hopkins Rd., Laurel, MD 20723. E-mail: christine.piatko@jhuapl.edu.

[||] Carnegie Technologies Institute, Carnegie Mellon University, Pittsburgh, PA 15213. E-mail: lsl@cs.cmu.edu.

[#] BNN Technologies, 10 Moulton Street, Cambridge, MA 02138. E-mail: smiller@bbn.com.

Submission received: 27 March 2011; revised submission received: 28 September 2011; accepted for publication: 30 November 2011.

This article describes the resource- and system-building efforts of an 8-week Johns Hopkins University Human Language Technology Center of Excellence Summer Camp for Applied Language Exploration (SCALE-2009) on Semantically Informed Machine Translation (SIMT). We describe a new modality/negation (MN) annotation scheme, the creation of a (publicly available) MN lexicon, and two automated MN taggers that we built using the annotation scheme and lexicon. Our annotation scheme isolates three components of modality and negation: a trigger (a word that conveys modality or negation), a target (an action associated with modality or negation), and a holder (an experiencer of modality). We describe how our MN lexicon was semi-automatically produced and we demonstrate that a structure-based MN tagger results in precision around 86% (depending on genre) for tagging of a standard LDC data set.

We apply our MN annotation scheme to statistical machine translation using a syntactic framework that supports the inclusion of semantic annotations. Syntactic tags enriched with semantic annotations are assigned to parse trees in the target-language training texts through a process of tree grafting. Although the focus of our work is modality and negation, the tree grafting procedure is general and supports other types of semantic information. We exploit this capability by including named entities, produced by a pre-existing tagger, in addition to the MN elements produced by the taggers described here. The resulting system significantly outperformed a linguistically naive baseline model (Hiero), and reached the highest scores yet reported on the NIST 2009 Urdu–English test set. This finding supports the hypothesis that both syntactic and semantic information can improve translation quality.

1. Introduction

This article describes the resource- and system-building efforts of an 8-week Johns Hopkins Human Language Technology Center of Excellence *Summer Camp for Applied Language Exploration* (SCALE-2009) on Semantically Informed Machine Translation (SIMT) (Baker et al. 2010a, 2010b, 2010c, 2010d). Specifically, we describe our modality/negation (MN) annotation scheme, a (publicly available) MN lexicon, and two automated MN taggers that were built using the lexicon and annotation scheme.

Our annotation scheme isolates three components of modality and negation: a trigger (a word that conveys modality or negation), a target (an action associated with modality or negation), and a holder (an experiencer of modality). Two examples of MN tagging are shown in Figure 1.

Note that modality and negation are unified into single MN tags (e.g., the "Able" modality tag is combined with "NOT" to form the "NOTAble" tag) and also that

(1) **Input:** Americans should know that we can not hand over Dr. Khan to them.
Output: Americans <TrigRequire should> <TargRequire know> that we <TrigAble can> <TrigNegation not> <TargNOTAble hand> over Dr. Khan to them.

(2) **Input:** He managed to hold general elections in the year 2002, but he can not be ignorant of the fact that the world at large did not accept these elections.
Output: He <TrigSucceed managed> to <TargSucceed hold> general elections in the year 2002, but he <TrigAble can> <TrigNegation not> <TargNOTAble be> ignorant of the fact that the world at large did <TrigNegation not> <TrigBelief accept> these <TargNOTBelief elections>.

Figure 1
Modality/negation tagging examples.

MN tags occur in pairs of triggers (e.g., TrigAble and TrigNegation) and targets (e.g., TargNOTAble).

We apply our modality and negation mechanism to the problem of Urdu–English machine translation using a technique that we call **tree grafting**. This technique incorporates syntactic labels and semantic annotations in a unified and coherent framework for implementing semantically informed machine translation. Our framework is not limited to the semantic annotations produced by the MN taggers that are the subject of this article and we exploit this capability to additionally include named-entity annotations produced by a pre-existing tagger. By augmenting hierarchical phrase-based translation rules with syntactic labels that were extracted from a parsed parallel corpus, and further augmenting the parse trees with markers for modality, negation, and entities (through the tree grafting process), we produced a better model for translating Urdu and English. The resulting system significantly outperformed the linguistically naive baseline Hiero model, and reached the highest scores yet reported on the NIST 2009 Urdu–English translation task.

We note that although our largest gains were from syntactic enrichments to the model, smaller (but significant) gains were achieved by injecting semantic knowledge into the syntactic paradigm. Verbal semantics (modality and negation) contributed slightly more gains than nominal semantics (named entities) and their combined gains were the sum of their individual contributions.

Of course, the limited semantic types we explored (modality, negation, and entities) are only a small piece of the much larger semantic space, but demonstrating success on these semantic aspects of language, the combination of which has been unexplored by the statistical machine translation community, bodes well for (larger) improvements based on the incorporation of other semantic aspects (e.g., relations and temporal knowledge). Moreover, we believe this syntactic framework to be well suited for further exploration of the impact of many different types of semantics on the quality of machine-translation (MT) output. Indeed, it would not have been possible to initiate the current study without the foundational work that gave rise to a syntactic paradigm that could support these semantic enrichments.

In the SIMT paradigm, semantic elements (e.g., modality/negation) are identified in the English portion of a parallel training corpus and projected to the source language (in our case, Urdu) during a process of syntactic alignment. These semantic elements are subsequently used in the translation rules that are extracted from the parallel corpus. The goal of adding them to the translation rules is to constrain the space of possible translations to more grammatical and more semantically coherent output. We explored whether including such semantic elements could improve translation output in the face of sparse training data and few source language annotations. Results were encouraging. Translation quality, as measured by the Bleu metric (Papineni et al. 2002), improved when the training process for the Joshua machine translation system (Li et al. 2009) used in the SCALE workshop included MN annotation.

We were particularly interested in identifying modalities and negation because they can be used to characterize events in a variety of automated analytic processes. Modalities and negation can distinguish realized events from unrealized events, beliefs from certainties, and can distinguish positive and negative instances of entities and events. For example, the correct identification and retention of negation in a particular language—such as a single instance of the word "not"—is very important for a correct representation of events and likewise for translation.

The next two sections examine related work and the motivation behind the SIMT approach. Section 4 defines the theoretical framework for our MN lexicon and automatic

MN taggers. Section 5 presents the MN annotation scheme used by our human annotators and describes the creation of a MN lexicon based on this scheme. Section 6 presents two types of MN taggers—one that is string-based and one that is structure-based—and evaluates the effectiveness of the structure-based tagger. Section 7 then presents implementation details of the semantically informed syntactic system and describes the results of its application. Finally, Section 8 presents conclusions and future work.

2. Related Work

The development of annotation schemes has become an area of computational linguistics development in its own right, often separate from machine learning applications. Many projects began as strictly linguistic projects that were later adapted for computational linguistics. When an annotation scheme is consistent and well developed, its subsequent application to NLP systems is most effective. For example, the syntactic annotation of parse trees in the Penn Treebank (Marcus, Marcinkiewicz, and Santorini 1993) had a tremendous effect on parsing and on Natural Language Processing in general.

In the case of semantic annotations, each tends to have its unique area of focus. Although the labeling conventions may differ, a layer of modality annotation over verb role annotation, for example, can have a complementary effect of providing more information, rather than being viewed as a competing scheme. We review some of the major semantic annotation efforts here.

Propbank (Palmer, Gildea, and Kingsbury 2005) is a set of annotations of predicate–argument structure over parse trees. First annotated as an overlay to the Penn Treebank, Propbank annotation now exists for other corpora. Propbank annotation aims to answer the question *Who did what to whom?* for individual predicates. It is tightly coupled with the behavior of individual verbs. FrameNet (Baker, Fillmore, and Lowe 1998), a frame-based lexical database that associates each word in the database with a semantic frame and semantic roles, is also associated with annotations at the lexical level. WordNet (Fellbaum 1998) is a very widely used online lexical taxonomy which has been developed in numerous languages. WordNet nouns, verbs, adjectives, and adverbs are organized into synonym sets. PropBank, FrameNet, and WordNet cover the word senses and argument-taking properties of many modal predicates.

The Prague Dependency Treebank (Hajič et al. 2001; Böhmová, Cinková, and Hajičová 2005) (PDT) is a multi-level system of annotation for texts in Czech and other languages, with its roots in the Prague school of linguistics. Besides a morphological layer and an analytical layer, there is a Tectogrammatical layer. The Tectogrammatical layer includes functional relationships, dependency relations, and co-reference. The PDT also integrates propositional and extra-propositional meanings in a single annotation framework.

The Penn Discourse Treebank (PDTB) (Webber et al. 2003; Prasad et al. 2008) annotates discourse connectives and their arguments over a portion of the Penn Treebank. Within this framework, senses are annotated for the discourse connectives in a hierarchical scheme. Relevant to the current work, one type of tag in the scheme is the Conditional tag, which includes hypothetical, general, unreal present, unreal past, factual present, and factual past arguments.

The PDTB work is related to that of Wiebe, Wilson, and Cardie (2005) for establishing the importance of attributing a belief or assertion expressed in text to its agent (equivalent to the notion of *holder* in our scheme). The annotation scheme is designed to capture the expression of opinions and emotions. In the PDTB, each discourse relation and its two arguments are annotated for attribution. The attribute features are the Source or agent, the Type (assertion propositions, belief propositions, facts, and eventualities), scopal polarity, and determinacy. Scopal polarity is annotated on relations and their arguments to identify cases when verbs of attribution are negated on the surface but the negation takes scope over the embedded clause. An example is the sentence "Having the dividend increases is a supportive element in the market outlook *but I don't think it's a main consideration*." Here, the second argument (the clause following *but*) is annotated with a "Neg" marker, meaning "I think it's not a main consideration."

Wilson, Wiebe, and Hoffman (2009) describe the importance of correctly interpreting polarity in the context of sentiment analysis, which is the task of identifying positive and negative opinions, emotions, and evaluations. The authors have established a set of features to distinguish between positive and negative polarity and discuss the importance of correctly analyzing the scope of the negation and the modality (e.g., whether the proposition is asserted to be real or not real).

A major annotation effort for temporal and event expressions is the TimeML specification language, which has been developed in the context of reasoning for question answering (Saurí, Verhagen, and Pustejovsky 2006). TimeML, which includes modality annotation on events, is the basis for creating the TimeBank and FactBank corpora (Pustejovsky et al. 2006; Saurí and Pustejovsky 2009). In FactBank, event mentions are marked with their degree of factuality.

Recent work incorporating modality annotation includes work on detecting certainty and uncertainty. Rubin (2007) describes a scheme for five levels of certainty, referred to as Epistemic modality, in news texts. Annotators identify explicit certainty markers and also take into account Perspective, Focus, and Time. Focus separates certainty into facts and opinions, to include attitudes. In our scheme, Focus would be covered by *want* and *belief* modality. Also, separating focus and uncertainty can allow the annotation of both on one trigger word. Prabhakaran, Rambow, and Diab (2010) describe a scheme for automatic committed belief tagging. Committed belief indicates the writer believes the proposition. The authors use a previously annotated corpus of committed belief, non-committed belief, and not applicable (Diab et al. 2009), and derive features for machine learning from parse trees. The authors desire to combine their work with FactBank annotation.

The CoNLL-2010 shared task (Farkas et al. 2010) was about the detection of cues for uncertainty and their scope. The task was described as "hedge detection," that is, finding statements which do not or cannot be backed up with facts. Auxiliary verbs such as *may*, *might*, *can*, and so forth, are one type of hedge cue. The training data for the shared task included the BioScope corpus (Szarvas et al. 2008), which is manually annotated with negation and speculation cues and their scope, and paragraphs from Wikipedia possibly containing hedge information. Our scheme also identifies cues in the form of triggers, but our desired outcome is to cover the full range of modalities and not just certainty and uncertainty. To identify scope, we use syntactic parse trees, as was allowed in the CoNLL task.

The textual entailment literature includes modality annotation schemes. Identifying modalities is important to determine whether a text entails a hypothesis. Bar-Haim et al. (2007) include polarity based rules and negation and modality annotation rules. The polarity rules are based on an independent polarity lexicon (Nairn, Condorovdi, and Karttunen 2006). The annotation rules for negation and modality of predicates are based on identifying modal verbs, as well as conditional sentences and modal adverbials. The authors read the modality off parse trees directly using simple structural rules for modifiers.

Earlier work describing the difficulty of correctly translating modality using machine translation includes Sigurd and Gawrónska (1994) and Murata et al. (2005). Sigurd and Gawrónska (1994) write about rule based frameworks and how using alternate grammatical constructions such as the passive can improve the rendering of the modal in the target language. Murata et al. (2005) analyze the translation of Japanese into English by several systems, showing they often render the present incorrectly as the progressive. The authors trained a support vector machine to specifically handle modal constructions, whereas our modal annotation approach is a part of a full translation system.

We now consider other literature, relating to tree-grafting and machine translation. Our tree-grafting approach builds on a technique used for tree augmentation in Miller et al. (2000), where parse-tree nodes are augmented with semantic categories. In that earlier work, tree nodes were augmented with relations, whereas we augmented tree nodes with modality and negation. The parser is subsequently retrained for both semantic and syntactic processing. The semantic annotations were done manually by students who were provided a set of guidelines and then merged with the syntactic trees automatically. In our work we tagged our corpus with entities, modality, and negation automatically and then grafted them onto the syntactic trees automatically, for the purpose of training a statistical machine translation system. An added benefit of the extracted translation rules is that they are capable of producing semantically tagged Urdu parses, despite the fact that the training data were processed by only an English parser and tagger.

Related work in syntax-based MT includes that of Huang and Knight (2006), where a series of syntax rules are applied to a source language string to produce a target language phrase structure tree. The Penn English Treebank (Marcus, Marcinkiewicz, and Santorini 1993) is used as the source for the syntactic labels and syntax trees are relabeled to improve translation quality. In this work, node-internal and node-external information is used to relabel nodes, similar to earlier work where structural context was used to relabel nodes in the parsing domain (Klein and Manning 2003). Klein and Manning's methods include lexicalizing determiners and percent markers, making more fine-grained verb phrase (VP) categories, and marking the properties of sister nodes on nodes. All of these labels are derivable from the trees themselves and not from an auxiliary source. Wang et al. (2010) use this type of node splitting in machine translation and report a small increase in BLEU score.

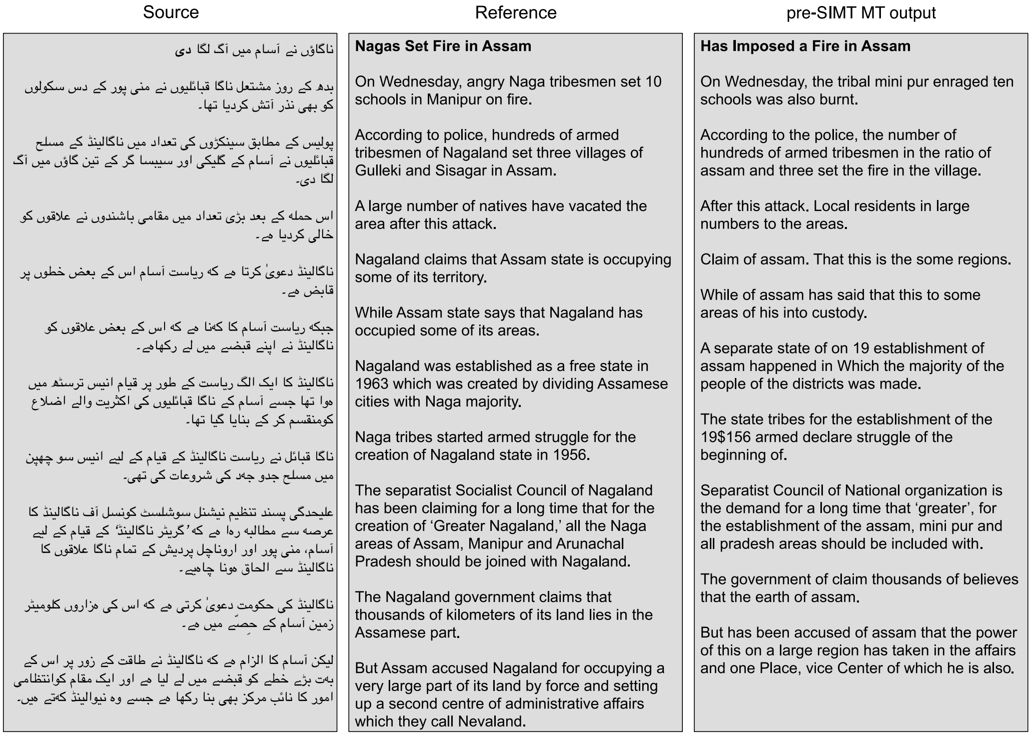
We use the methods described in Zollmann and Venugopal (2006) and Venugopal, Zollmann, and Vogel (2007) to induce synchronous grammar rules, a process which requires phrase alignments and syntactic parse trees. Venugopal, Zollmann, and Vogel (2007) use generic non-terminal category symbols, as in Chiang (2005), as well as grammatical categories from the Stanford parser (Klein and Manning 2003). Their method for rule induction generalizes to any set of non-terminals. We further refine this process by adding semantic notations onto the syntactic non-terminals produced by a Penn Treebank trained parser, thus making the categories more informative.

In the parsing domain, the work of Petrov and Klein (2007) is related to the current work. In their work, rule splitting and rule merging are applied to refine parse trees during machine learning. Hierarchical splitting leads to the creation of learned categories that have linguistic relevance, such as a breakdown of a determiner category into two subcategories of determiners by number, that is, *this* and *that* group together as do *some* and *these*. We augment parse trees by category insertion in cases where a semantic category is inserted as a node in a parse tree, after the English side of the corpus has been parsed by a statistical parser.

3. SIMT Motivation

As in many of the frameworks described herein, the aim of the SIMT effort was to provide a generalized framework for representing structured semantic information, such as modality and negation. Unlike many of the previous semantic annotation efforts (where the emphasis tends to be on English), however, our approach is designed to be directly integrated into a translation engine, with the goal of translating highly divergent language pairs, such as Urdu and English. As such, our choice of annotation scheme—illustrated in the trigger-target example shown in Figure 1—was based on a simplified structural representation that is general enough to accommodate divergent modality/negation phenomena, easy for language experts to follow, and straightforward to integrate into a tree-grafting mechanism for MT. Our objective is to investigate whether incorporating this sort of information into machine translation systems could produce better translations, particularly in settings where only small parallel corpora are available.

It is informative to look at an example translation to understand the challenges of translating important semantic elements when working with a low-resource language pair. Figure 2 shows an example taken from the 2008 NIST Urdu–English translation task, and illustrates the translation quality of a state-of-the-art Urdu–English system (prior to the SIMT effort). The small amount of training data for this language pair (see



Source	Reference	pre-SIMT MT output
ناگاؤں نے آسام میں آگ لگا دی	**Nagas Set Fire in Assam**	**Has Imposed a Fire in Assam**
بدھ کے روز مشتعل ناگا قبائلیوں نے منی پور کے دس سکولوں کو بھی نذر آتش کردیا تھا۔	On Wednesday, angry Naga tribesmen set 10 schools in Manipur on fire.	On Wednesday, the tribal mini pur enraged ten schools was also burnt.
پولیس کے مطابق سینکڑوں کی تعداد میں ناگالینڈ کے مسلح قبائلیوں نے آسام کے گلیکی اور سیبسا گر کے تین گاؤں میں آگ لگا دی۔	According to police, hundreds of armed tribesmen of Nagaland set three villages of Gulleki and Sisagar in Assam.	According to the police, the number of hundreds of armed tribesmen in the ratio of assam and three set the fire in the village.
اس حملہ کے بعد بڑی تعداد میں مقامی باشندوں نے علاقوں کو خالی کردیا ہے۔	A large number of natives have vacated the area after this attack.	After this attack. Local residents in large numbers to the areas.
ناگالینڈ دعویٰ کرتا ہے کہ ریاست آسام اس کے بعض خطوں پر قابض ہے۔	Nagaland claims that Assam state is occupying some of its territory.	Claim of assam. That this is the some regions.
جبکہ ریاست آسام کا کہنا ہے کہ اس کے بعض علاقوں کو ناگالینڈ نے اپنے قبضے میں لے رکھاہے۔	While Assam state says that Nagaland has occupied some of its areas.	While of assam has said that this to some areas of his into custody.
ناگالینڈ کا ایک الگ ریاست کے طور پر قیام انیس ترسٹھ میں ہوا تھا جسے آسام کے ناگا قبائلیوں کی اکثریت والے اضلاع کومنقسم کر کے بنایا گیا تھا۔	Nagaland was established as a free state in 1963 which was created by dividing Assamese cities with Naga majority.	A separate state of on 19 establishment of assam happened in Which the majority of the people of the districts was made.
ناگا قبائل نے ریاست ناگالینڈ کے قیام کے لیے انیس سو چھپن میں مسلح جدو جہد کی شروعات کی تھی۔	Naga tribes started armed struggle for the creation of Nagaland state in 1956.	The state tribes for the establishment of the 19$156 armed declare struggle of the beginning of.
علیحدگی پسند تنظیم نیشنل سوشلسٹ کونسل آف ناگالینڈ کا عرصہ سے مطالبہ رہا ہے کہ ’گریٹر ناگالینڈ‘ کے قیام کے لیے آسام، منی پور اور اروناچل پردیش کے تمام ناگا علاقوں کا ناگالینڈ سے الحاق ہونا چاہیے۔	The separatist Socialist Council of Nagaland has been claiming for a long time that for the creation of 'Greater Nagaland,' all the Naga areas of Assam, Manipur and Arunachal Pradesh should be joined with Nagaland.	Separatist Council of National organization is the demand for a long time that 'greater', for the establishment of the assam, mini pur and all pradesh areas should be included with.
ناگالینڈ کی حکومت دعویٰ کرتی ہے کہ اس کی ہزاروں کلومیٹر زمین آسام کے حِصّے میں ہے۔	The Nagaland government claims that thousands of kilometers of its land lies in the Assamese part.	The government of claim thousands of believes that the earth of assam.
لیکن آسام کا الزام ہے کہ ناگالینڈ نے طاقت کے زور پر اس کے بہت بڑے خطے کو قبضے میں لے لیا ہے اور ایک مقام کوانتظامی امور کا نائب مرکز بھی بنا رکھا ہے جسے وہ نیوالینڈ کہتے ہیں۔	But Assam accused Nagaland for occupying a very large part of its land by force and setting up a second centre of administrative affairs which they call Nevaland.	But has been accused of assam that the power of this on a large region has taken in the affairs and one Place, vice Center of which he is also.

Figure 2
An example of Urdu–English translation. Shown are an Urdu source document, a reference translation produced by a professional human translator, and MT output from a phrase-based model (Moses) without linguistic information, which is representative of state-of-the-art MT quality before the SIMT effort.

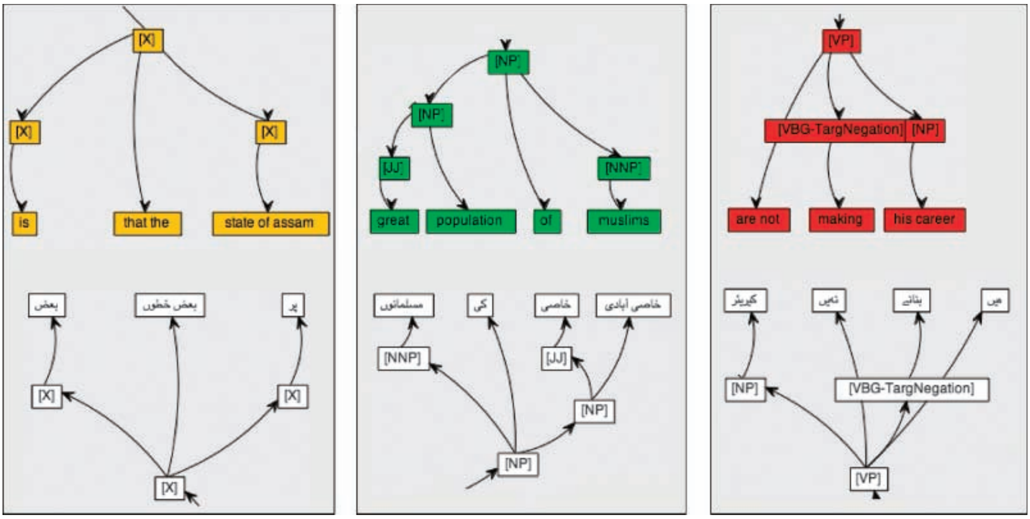
Table 1
The size of the various data sets used for the experiments in this article including the training, development (dev), incremental test set (devtest), and blind test set (test). The dev/devtest was a split of the NIST08 Urdu–English test set, and the blind test set was NIST09.

		Urdu		English	
set	lines	tokens	types	tokens	types
training	202k	1.7M	56k	1.7M	51k
dev	981	21k	4k	19k	4k
devtest	883	22k	4k	19–20k	4k
test	1,792	42k	6k	38–41k	5k

Table 1) results in significantly degraded translation quality compared, for example, to an Arabic–English system that has more than 100 times the amount of training data.

The output in Figure 2 was produced using Moses (Koehn et al. 2007), a state-of-the-art phrase-based MT system that by default does not incorporate any linguistic information (e.g., syntax or morphology or transliteration knowledge). As a result, words that were not directly observed in the bilingual training data were untranslatable. Names, in particular, are problematic. For example, the lack of translation for *Nagaland* and *Nagas* induces multiple omissions throughout the translated text, thus producing several instances where the *holder* of a claim (or *belief*) is missing. This is because out-of-vocabulary words are deleted from the Moses output.

We use syntactic and semantic tags as higher-order symbols inside the translation rules used by the translation models. Generic symbols in translation rules (i.e., the non-terminal symbol "X") were replaced with structured information at multiple levels of abstraction, using a tree-grafting approach that we describe subsequently. Figure 3



Standard Hierarchical Rules Syntactic Enhancements Semantically Informed Rules

Figure 3
The evolution of a semantically informed approach to our synchronous context-free grammars. At the start of the 8 weeks the decoder used translation rules with a single generic non-terminal symbol. Later syntactic categories were used, and by the end of the workshop the translation rules included semantic elements such as modalities and negation, as well as named entities.

illustrates the evolution of the translation rules that we used, first replacing "X" with grammatical categories and then with categories corresponding to semantic units.

The semantic units that we examined in this effort were modalities and negation (indications that a statement represents something that has/hasn't taken place or is/isn't a belief or an intention) and named entities (such as people or organizations). Other semantic units, such as relations between entities and events, were not part of this effort but we believe they could be similarly incorporated into the framework. We chose to examine semantic units that canonically exhibit two different syntactic types: verbal, in the case of modality and negation, and nominal, in the case of named entities.

Although used in this effort, named entities were not the focus of our research efforts in SIMT. Rather, we focused on the development of an annotation scheme for modality and negation and its use in MT, while relying on a pre-existing hidden Markov model (HMM)-based tagger derived from Identifinder (Bikel, Schwartz, and Weischedel 1999) to produce entity tags. Thus, the remainder of this article will focus on our MN annotation scheme, two MN taggers produced by the effort, and on the integration of semantics in the SIMT paradigm.

4. Modality and Negation

Modality is an extra-propositional component of meaning. In *John may go to NY*, the basic proposition is *John go to NY* and the word *may* indicates modality and is called the **trigger** in our work. van der Auwera and Amman (2005) define core cases of modality: *John must go to NY* (epistemic necessity), *John might go to NY* (epistemic possibility), *John has to leave NY now* (deontic necessity), and *John may leave NY now* (deontic possibility). Larreya (2009) defines the core cases slightly differently as *root* and *epistemic*. Root modality in Larreya's taxonomy includes physical modality (*He had to stop. The road was blocked*) and deontic modality (*You have to stop*). Epistemic modality includes problematic modality (*You must be tired*) and implicative modality (*You have to be mad to do that*). Many semanticists (Kratzer 1991, von Fintel and Iatridou 2006) define modality as quantification over possible worlds. *John might leave NY* means that there exist some possible worlds in which John leaves NY. Another view of modality relates more to a speaker's attitude toward a proposition (McShane, Nirenburg, and Zacharski).

We incorporate negation as an inextricably intertwined component of modality, using the term "modality/negation (MN)" to refer to our resources (lexicons) and processes (taggers). We adopt the view that modality includes several types of attitudes that a speaker might have (or not have) toward an event or state. From the point of view of the reader or listener, modality might indicate factivity, evidentiality, or sentiment. Factivity is related to whether an event, state, or proposition happened or didn't happen. It distinguishes things that happened from things that are desired, planned, or probable. Evidentiality deals with the source of information and may provide clues to the reliability of the information. Did the speaker have first-hand knowledge of what he or she is reporting, or was it hearsay or inferred from indirect evidence? Sentiment deals with a speaker's positive or negative feelings toward an event, state, or proposition.

Our project was limited to modal words and phrases—and their negations—that are related to factivity. Beyond the core cases of modality, however, we include some aspects of speaker attitude such as intent and desire. We included these because they are often not separable from the core cases of modality. For example, *He had to go* may include the ideas that someone wanted him to go, that he might not have wanted to go, that at some point after coercion he intended to go, and that at some point he was able to go (Larreya 2009).

Our focus was on the eight modalities in Figure 4, where P is a proposition (the *target* of the *triggering* modality) and H is the holder (experiencer or cognizer of the modality). Some of the eight factivity-related modalities may overlap with sentiment or evidentiality. For example, *want* indicates that the proposition it scopes over may not be a fact (it may just be desired), but it also expresses positive sentiment toward the proposition it scopes over. We assume that sentiment and evidentiality are covered under separate coding schemes, and that words like *want* would have two tags, one for sentiment and one for factivity.

5. The Modality/Negation Annotation Scheme

The challenge of creating an MN annotation scheme was to deal with the complex scoping of modalities with each other and with negation, while at the same time creating a simplified operational procedure that could be followed by language experts without special training. Here we describe our MN annotation framework, including a set of linguistic simplifications, and then we present our methodology for creation of a publicly available MN lexicon. The modality annotation scheme is fully documented in a set of guidelines that were written with English example sentences (Baker et al. 2010c). The guidelines can be used to derive hand-tagged evaluation data for English and they also include a section that contains a set of Urdu trigger-word examples.

During the SCALE workshop, some Urdu speakers used the guidelines to annotate a small corpus of Urdu by hand, which we reserved for future work. The Urdu corpus could be useful as an evaluation corpus for automatically tagged Urdu, such as one derived from rule projection in the Urdu–English MT system, a method we describe further in Section 7. Also, although we did not annotate a very large Urdu corpus, more data could be manually annotated to train an automatic Urdu tagger in the future.

5.1 Anatomy of Modality/Negation in Sentences

In sentences that express modality, we identify three components: a trigger, a target, and a holder. The **trigger** is the word or string of words that expresses modality or negation. The **target** is the event, state, or relation over which the modality scopes. The **holder** is

- **Requirement:** does H require P?
- **Permissive:** does H allow P?
- **Success:** does H succeed in P?
- **Effort:** does H try to do P?
- **Intention:** does H intend P?
- **Ability:** can H do P?
- **Want:** does H want P?
- **Belief**: with what strength does H believe P?

Figure 4
Eight modalities used for tagging. H = the holder of the modality; P = the proposition over which the modality has scope.

the experiencer or cognizer of the modality. The trigger can be a word such as *should*, *try*, *able*, *likely*, or *want*. It can also be a negative element such as *not* or *n't*. Often, modality or negation is expressed without a lexical trigger. For a typical declarative sentence (e.g., *John went to NY*), the default modality is strong belief when no lexical trigger is present. Modality can also be expressed constructionally. For example, Requirement can be expressed in Urdu with a dative subject and infinitive verb followed by a verb that means to happen or befall.

5.2 Linguistic Simplifications for Efficient Operationalization

Six linguistic simplifications were made for the sake of efficient operationalization of the annotation task. The first linguistic simplification deals with the scope of modality and negation. The first given sentence indicates scope of modality over negation. The second sentence indicates scope of negation over modality:

- He tried not to criticize the president.
- He didn't try to criticize the president.

The interaction of modality with negation is complex, but was operationalized easily in the menu of 13 choices shown in Figure 5. First consider the case where negation scopes over modality. Four of the 13 choices are composites of negation scoping over modality. For example, the annotators can choose *try* or *not try* as two separate modalities. Five modalities (Require, Permit, Want, Firmly Believe, and Believe) do not have a negated form. For three of these modalities (Want, Firmly Believe, and Believe), this is because they are often transparent to negation. For example, *I do not believe that he left NY* sometimes means the same as *I believe he didn't leave NY*. Merging the two is obviously a simplification, but it saves the annotators from having to make a difficult decision.

- H requires [P to be true/false]
- H permits [P to be true/false]
- H succeeds in [making P true/false]
- H does not succeed in [making P true/false]
- H is trying [to make P true/false]
- H is not trying [to make P true/false]
- H intends [to make P true/false]
- H does not intend [to make P true/false]
- H is able [to make P true/false]
- H is not able [to make P true/false]
- H wants [P to be true/false]
- H firmly believes [P is true/false]
- H believes [P may be true/false]

Figure 5
Thirteen menu choices for Modality/Negation annotation. H = the holder of the modality; P = the proposition over which the modality has scope.

The second linguistic simplification is related to a duality in meaning between *require* and *permit*. Not requiring P to be true is similar in meaning to permitting P to be false. Thus, annotators were instructed to label *not require P to be true* as *Permit P to be false*. Conversely, *not Permit P to be true* was labeled as *Require P to be false*.

After the annotator chooses the modality, the scoping of modality over negation takes place as a second decision. For example, for the sentence *John tried not to go to NY*, the annotator first identifies *go* as the target of a modality and then chooses *try* as the modality. Finally, the annotator chooses *false* as the polarity of the target.

The third simplification relates to entailments between modalities. Many words have complex meanings that include components of more than one modality. For example, if one managed to do something, one tried to do it and one probably wanted to do it. Thus, annotators were provided a specificity-ordered modality list as in Figure 5, and were asked to choose the first applicable modality. We note that this list corresponds to two independent "entailment groupings," ordered by specificity:

- {*requires* → *permits*}
- {*succeeds* → *tries* → *intends* → *is able* → *wants*}

Inside the entailment groupings, the ordering corresponds to an entailment relation: For example, *succeeds* can only occur if *tries* has occurred. Also, the {*requires* → ... } entailment grouping is taken to be more specific than (ordered before) the {*succeeds* → ... } entailment grouping. Moreover, both entailment groupings are taken to be more specific than *believes*, which is not in an entailment relation with any of the other modalities.

The fourth simplification, already mentioned, is that sentences without an overt trigger word are tagged as *firmly believes*. This heuristic works reasonably well for the types of documents we were working with, although one could imagine genres such as fiction in which many sentences take place in an alternate possible world (imagined, conditional, or counterfactual) without explicit marking.

The fifth linguistic simplification is that we did not require annotators to mark nested modalities. For a sentence like *He might be able to go to NY* the target word *go* is marked as ability, but *might* is not annotated for Belief modality. This decision was based on time limits on the annotation task; there was not enough time for annotators to deal with syntactic scoping of modalities over other modalities.

Finally, we did not mark the holder H because of the short time frame for workshop preparation. We felt that identifying the triggers and targets would be most beneficial in the context of machine translation.

5.3 The English Modality/Negation Lexicon

Using the given framework, we created an MN lexicon that was incorporated into an MN tagging scheme to be described in Section 6. Entries in the MN lexicon consist of: (1) A string of one or more words: for example, *should* or *have need of*. (2) A part of speech for each word: The part of speech helps us avoid irrelevant homophones such as the noun *can*. (3) An MN designator: one of the 13 modality/negation cases described previously. (4) A head word (or *trigger*): the primary phrasal constituent to cover cases where an entry is a multi-word unit (e.g., the word *hope* in *hope for*). (5) One or more subcategorization codes derived from the Longman Dictionary of Contemporary English (LDOCE).

We produced the full English MN lexicon semi-automatically. First, we gathered a small seed list of MN trigger words and phrases from our modality annotation manual (Baker et al. 2010c). Then, we expanded this small list of MN trigger words by running an on-line search for each of the words, specifically targeting free on-line thesauri (e.g., `thesaurus.com`), to find both synonymous and antonymous words. From these we manually selected the words we thought triggered modality (or their corresponding negative variants) and filtered out words that we thought didn't trigger modality. The resulting list of MN trigger words and phrases contained about 150 lemmas.

We note that most intransitive (LDOCE) codes were not applicable to modality/negation constructions. For example, *hunger* (in the *Want* modality class) has a modal reading of "desire" when combined with the preposition *for* (as in *she hungered for a promotion*), but we do not consider it to be modal when it is used in the somewhat archaic sentence *He hungered*, meaning that he did not have enough to eat. Thus the LDOCE code `I` associated with the verb *hunger* was hand-changed to `I-FOR`. There were 43 such cases. Once the LDOCE codes were hand-verified (and modified accordingly), the mapping to subcategorization codes was applied.

The MN lexicon is publicly available at `http://www.umiacs.umd.edu/∼bonnie/ModalityLexicon.txt`. An example of an entry is given in Figure 6, for the verb *need*.

6. Automatic Modality/Negation Annotation

An MN tagger produces text or structured text in which modality or negation triggers and/or targets are identified. Automatic identification of the holders of modalities was beyond the scope of our project because the holder is often not explicitly stated in the sentence in which the trigger and target occur. This section describes two types of MN taggers—one that is string-based and one that is structure-based.

6.1 The String-Based English Modality/Negation Tagger

The string-based tagger operates on text that has been tagged with parts of speech by a Collins-style statistical parser (Miller et al. 1998). The tagger marks spans of words/phrases that exactly match MN trigger words in the MN lexicon described previously, and that exactly match the same parts of speech. This tagger identifies the target of each modality/negation using the heuristic of tagging the next non-auxiliary verb to the right of the trigger. Spans of words can be tagged multiple times with different types of triggers and targets.

String:	Need
Pos:	VB
Modality:	Require
Trigger:	Need
Subcat:	**V3-passive-basic** – More citizens are needed to vote.
Subcat:	**V3-I3-basic** – The government will need to work continuously for at least a year. We will need them to work continuously.
Subcat:	**T1-monotransitive-for-V3-verbs** – We need a Sir Sayyed again to maintain this sentiment.
Subcat:	**T1-passive-for-V3-verb** – Tents are needed.
Subcat:	**Modal-auxiliary-basic** – He need not go.

Figure 6
Modality lexicon entry for *need*.

We found the string-based MN tagger to produce output that matched about 80% of the sentence-level tags produced by our structure-based tagger, the results of which are described next. Although string-based tagging is fast and reasonably accurate in practice, we opted to focus on the indepth analysis of modality/negation of our SIMT results using the more accurate structure-based tagger.

6.2 The Structure-Based English Modality/Negation Tagger

The structure-based MN tagger operates on text that has been parsed (Miller et al. 1998). We used a version of the parser that produces flattened trees. In particular, the flattener deletes VP nodes that are immediately dominated by VP or S and noun phrase (NP) nodes that are immediately dominated by PP or NP. The parsed sentences are processed by TSurgeon rules. Each TSurgeon rule consists of a pattern and an action. The pattern matches part of a parse tree and the action alters the parse tree. More specifically, the pattern finds an MN trigger word and its target and the action inserts tags such as `TrigRequire` and `TargRequire` for triggers and targets for the modality Require. Figure 7 shows output from the structure-based MN tagger. (Note that the sentence is disfluent: *Pakistan which could not reach semi-final, in a match against South African team for the fifth position Pakistan defeated South Africa by 41 runs.*) The example shows that *could* is a trigger for the Ability modality and *not* is a trigger for negation. *Reach* is a target for both Ability and Negation, which means that it is in the category of "H is not able [to make P true/false]" in our coding scheme. *Reach* is also a trigger for the Succeed modality and *semi-final* is its target.

The TSurgeon patterns are automatically generated from the verb class codes in the MN lexicon along with a set of 15 templates. Each template covers one situation such as the following: the target is the subject of the trigger; the target is the direct object of the trigger; the target heads an infinitival complement of the trigger; the target is a noun modified by an adjectival trigger, and so on. The verb class codes indicate

```
(TOP
 (S
  (NP
   (NNP Pakistan)
   (SBAR (WDT which)
    (S (MD TrigAble could)
       (RB TrigNegation not)
       (VB B TargAble TrigSucceed
        TargNegation reach)
       (ADJP
        (JJ TargSucceed semi-final))
        (, ,)
       (PP (IN in) (DT a)
           (NN match) (PP (IN against)
           (ADJP (JJ South) (JJ African))
            (NN team))
           (PP (IN for) (DT the)
               (JJ fifth) (NN position))
           (NP (NNP Pakistan))))))
  (VB D defeated)
  (NP (NNP South) (NNP Africa))
  (PP (IN by) (CD 41) (NNS runs)) (. .)))
```

Figure 7
Sample output from the structure-based MN tagger.

which templates are applicable for each trigger word. For example, a trigger verb in the transitive class may use two target templates, one in which the trigger is in active voice and the target is a direct object (*need tents*) and one in which the trigger is in passive voice and the target is a subject (*tents are needed*).

In developing the TSurgeon rules, we first conducted a corpus analysis for 40 of the most common trigger words in order to identify and debug the most broadly applicable templates. We then used LDOCE to assign verb classes to the remaining verbal triggers in the MN lexicon, and we associated one or more debugged templates with each verb class. In this way, the initial corpus work on a limited number of trigger words was generalized to a longer list of trigger words. Because the TSurgeon patterns are tailored to the flattened structures produced by our parser, it is not easily ported to new parser outputs. The MN lexicon itself is portable, however. Switching parsers would entail writing new TSurgeon templates, but the trigger words in the MN lexicon would still be automatically assigned to templates based on their verb classes.

The following example shows an example of a TSurgeon pattern–action pair for a sentence like *They were required to provide tents*. The pattern–action pair is intended to be used after a pre-processing stage in which labels such as "VoicePassive" and "AUX" have been assigned. "VoicePassive" is inserted by a pre-processing TSurgeon pattern because, in some cases, the target of a passive modality trigger word is in a different location from the target of the corresponding active modality trigger word. "AUX" is inserted during pre-processing to distinguish auxiliary uses of *have* and *be* from their uses as main verbs. The pattern portion of the pattern–action pair matches a node with label VB that is not already tagged as a trigger and that is passive and dominates the string "required". The VB node is also a sister to an S node, and the S node dominates a VB that is not an auxiliary (*provide* in this case). The action portion of the pattern–action pair inserts the string "TargReq" as the second daughter of the second VB and inserts the string "TrigReq" as the second daughter of the first VB.

```
VB=trigger !< /^Trig/ < VoicePassive < required  $..
   (S < (VB=target !< AUX))
insert (TargReq) >2 target
insert (TrigReq) >2 trigger
```

Verb-specific patterns such as this one were generalized in order to gain coverage of the whole modality lexicon. The specific lexical item, *required*, was replaced with a variable, as were the labels "TrigReq" and "TargReq." The pattern was then given a name, V3-passive-basic, where V3 is a verb class tag from LDOCE (described in Section 5.3) for verbs that take infinitive complements. We then looked up the LDOCE verb class labels for all of the verbs in the modality lexicon. Using this information, we could then generate a set of new, verb-specific patterns for each V3 verb in the modality lexicon.

6.3 Evaluating the Effectiveness of Structure-Based MN Tagging

We performed a manual inspection of the structure-based tagging output. We calculated precision by examining 229 instances of modality triggers that were tagged by our tagger from the English side of the NIST 09 MTEval training sentences. We analyzed precision in two steps, first checking for the correct syntactic position of the target and then checking the semantic correctness of the trigger and target. For 192 of the 229 triggers (around 84%), the targets were tagged in the correct syntactic location.

For example, for the sentence *A solution must be found to this problem* shown in Figure 8, the word *must* is a modality trigger word, and the correct target is the first

```
(S (NP (DT A) (NN solution))
   (VP (MD-TrigBelief must)
       (VP (VB be)
           (VP (VBN-TargBelief found)
               (PP (TO to) (NP (DT this) (NN problem)))))))
```

Figure 8
Example of embedded target head *found* inside VP *must be found*.

non-auxiliary verb heading a verb phrase that is contained in the syntactic complement of *must*. The syntactic complement of *must* is the verb phrase *be found to this problem*. The syntactic head of that verb phrase, *be*, is skipped because it is an auxiliary verb. The correct (embedded) target *found* is the head of the syntactic complement of *be*.

The 192 modality instances with structurally correct targets do not all have semantically correct tags. In this example, *must* is tagged as `TrigBelief`, where the correct tag would be `TrigRequire`. Also, because the MN lexicon was used without respect to word sense, words were sometimes erroneously identified as triggers. This includes non-modal uses of *work* (work with refugees), *reach* (reach a destination), and *attack* (attack a physical object), in constrast to modal uses of these words: *work for peace* (effort), *reach a goal* (succeed), and *attack a problem* (effort). Fully correct tagging of modality would need to include word sense disambiguation.

For 37 of the 229 triggers we examined, a target was not tagged in the correct syntactic position. In 12 of 37 incorrectly tagged instances the targets are inside compound nouns or coordinate structures (NP or VP), which are not yet handled by the modality tagger. The remaining 25 of the 37 incorrectly tagged instances had targets that were lost because the tagger does not yet handle all cases of nested modalities. Nested modalities occur in sentences like *They did not want to succeed in winning* where the target words *want* and *succeed* are also modality trigger words. Proper treatment of nested modalities requires consideration of scope and compositional semantics.

Nesting was treated in two steps. First, the modality tagger marked each word as a trigger and/or target. In *They did not want to succeed in winning*, *not* is marked as a trigger for negation, *want* is marked as a target of negation and a trigger of wanting, *succeed* is marked as a trigger of succeeding and a target of wanting, and *win* is marked as a target of succeeding. The second step in the treatment of nested modalities occurs during tree grafting, where the meanings of the nested modalities are composed. The tree grafting program correctly composes some cases of nested modalities. For example, the tag `TrigAble` composed with `TrigNegation` results in the target tag `TargNOTAble`, as shown in Figure 9. In other cases, where compositional semantics are not yet accommodated, the tree grafting program removed target labels from the trees, and those cases were counted as incorrect for the purpose of this evaluation.

```
(S
  (NP (EX there))
  (VP (VBZ is)
      (NP (NP (DT no) (NN difficulty))
          (SBAR (WHNP (WDT which))
            (S (VP (MD-TrigAble can)
                   (RB-TrigNegation not)
                   (VP (VB be)
                       (VP-TargNOTAble (VBN-TargNOTAble solved)))))))))
```

Figure 9
Example of modality composed with negation: TrigAble and TrigNegation combine to form NOTAble.

In the 229 instances that we examined, there were 14 in which a light verb or noun was the correct syntactic target, but not the correct semantic target. *Decision* would be a better target than *taken* in *The decision* **should** *be* **taken** *on delayed cases on the basis of merit.* We counted sentences with semantically light targets as correct in our evaluation because our goal was to identify the syntactic head of the target. The semantics of the target is a general issue, and we often find lexico-syntactic fluff between the trigger and the most semantically salient target in sentences like *We succeeded in our goal of winning the war* where "success in war" is the salient meaning.

With respect to recall, the tagger primarily missed special forms of negation in noun phrases and prepositional phrases: *There was* **no** *place to seek shelter*; *The buildings should be reconstructed,* **not** *with RCC, but with the wood and steel sheets.* More complex constructional and phrasal triggers were also missed: *President Pervaiz Musharraf has said that he will* **not rest unless** *the process of rehabilitation is completed.* Finally, we discovered some omissions from our MN lexicon: *It is not* **possible** *in the middle of winter to re-open the roads.* Further annotation experiments are planned, which will be analyzed to close such gaps and update the lexicon as appropriate.

Providing a quantitative measure of recall was beyond the scope of this project. At best we could count instances of sentences containing trigger words that were not tagged. We are also aware of many cases of modality that were not covered such as the modal uses of the future tense auxiliary *will* as in *That'll be John* (conjecture), *I'll do the dishes* (volition), *He won't do it* (non-volition), and *It will accommodate five* (ability) (Larreya 2009). Because of the complexity and subtlety of modality and negation, however, it would be impractical to count every clause (such as the *not rest unless* clause above) that had a nuance of non-factivity.

7. Semantically Informed Syntactic MT

This section describes the incorporation of our structured-based MN tagging into an Urdu–English machine-translation system using *tree grafting* for combining syntactic symbols with semantic categories (e.g., modality/negation). We note that a de facto Urdu MN tagger resulted from identifying the English MN trigger and target words in a parallel English–Urdu corpus, and then projecting the trigger and target labels to the corresponding words in Urdu syntax trees.

7.1 Refinement of Translation Grammars with Semantic Categories

We used synchronous context-free grammars (SCFGs) as the underlying formalism for our statistical models of translation. SCFGs provide a convenient and theoretically grounded way of incorporating linguistic information into statistical models of translation, by specifying grammar rules with syntactic non-terminals in the source and target languages. We refine the set of non-terminal symbols so that they not only include syntactic categories, but also semantic categories.

Chiang (2005) re-popularized the use of SCFGs for machine translation, with the introduction of his hierarchical phrase-based machine translation system, Hiero. Hiero uses grammars with a single non-terminal symbol "X" rather than using linguistically informed non-terminal symbols. When moving to linguistic grammars, we use Syntax Augmented Machine Translation (SAMT) developed by Venugopal, Zollmann, and Vogel (2007). In SAMT the "X" symbols in translation grammars are replaced with nonterminal categories derived from parse trees that label the English side of the

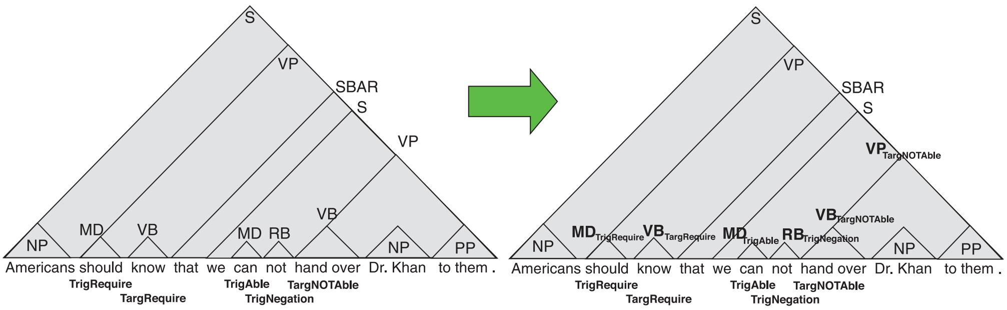


Figure 10
A sentence on the English side of the bilingual parallel training corpus is parsed with a syntactic parser, and also tagged with our modality tagger. The tags are then *grafted* onto the syntactic parse tree to form new categories like VP-TargNOTAble and VP-TargRequire. Grafting happens prior to extracting translation rules, which happens normally except for the use of the augmented trees.

Urdu–English parallel corpus.[1] We refine the syntactic categories by combining them with semantic categories. Recall that this progression was illustrated in Figure 3.

We extracted SCFG grammar rules containing modality, negation, and named entities using an extraction procedure that requires parse trees for one side of the parallel corpus. Although it is assumed that these trees are labeled and bracketed in a syntactically motivated fashion, the framework places no specific requirement on the label inventory. We take advantage of this characteristic by providing the rule extraction algorithm with augmented parse trees containing syntactic labels that have semantic annotations grafted onto them so that they additionally express semantic information.

Our strategy for producing semantically grafted parse trees involves three steps:

1. The English sentences in the parallel training data are parsed with a syntactic parser. In our work, we used the lexicalized probabilistic context free grammar parser provided by Basis Technology Corporation.
2. The English sentences are MN-tagged by the system described herein and named-entity-tagged by the Phoenix tagger (Richman and Schone 2008).
3. The modality/negation and entity markers are grafted onto the syntactic parse trees using a tree-grafting procedure. The grafting procedure was implemented as part of the SIMT effort. Details are further spelled out in Section 7.2.

Figure 10 illustrates how modality tags are grafted onto a parse tree. Note that although we focus the discussion here on the modality and negation, our framework is general and we were able to incorporate other semantic elements (specifically, named entities) into the SIMT effort.

Once the semantically grafted trees have been produced for the parallel corpus, the trees are presented, along with word alignments (produced by the Berkeley aligner), to the rule extraction software to extract synchronous grammar rules that are both

[1] For non-constituent phrases, composite CCG-style categories are used (Steedman 1999).

syntactically and semantically informed. These grammar rules are used by the decoder to produce translations. In our experiments, we used the Joshua decoder (Li et al. 2009), the SAMT grammar extraction software (Venugopal and Zollmann 2009), and special purpose-built tree-grafting software.

Figure 11 shows example semantic rules that are used by the decoder. The verb phrase rules are augmented with modality and negation, taken from the semantic categories listed in Table 2. Because these get marked on the Urdu source as well as the English translation, semantically enriched grammars also act as very simple named entity or MN taggers for Urdu. Only entities, modality, and negation that occurred in the parallel training corpus are marked in the output, however.

7.2 Tree-Grafting Algorithm

The overall scheme of our tree-grafting algorithm is to match semantic tags to syntactic categories. There are two inputs to the process. Each is derived from a common text file of sentences. The first input is a list of standoff annotations for the semantically tagged word sequences in the input sentences, indexed by sentence number. The second is a list of parse trees for the sentences in Penn Treebank format, indexed by sentence number.

Table 2 lists the modality/negation types that were produced by the MN tagger. For example, the sentence *The students are able to swim* is tagged as *The students are ⟨TrigAble⟩ to ⟨TargAble swim⟩*. The distinction between “Negation” and “NOT” corresponds to the difference between negation that is inherently expressed in the triggering lexical item and negation that is expressed explicitly as a separate lexical item. Thus, *I achieved my goal* is tagged “Succeed” and *I did not achieve my goal* is tagged as “NOTSucceed,”

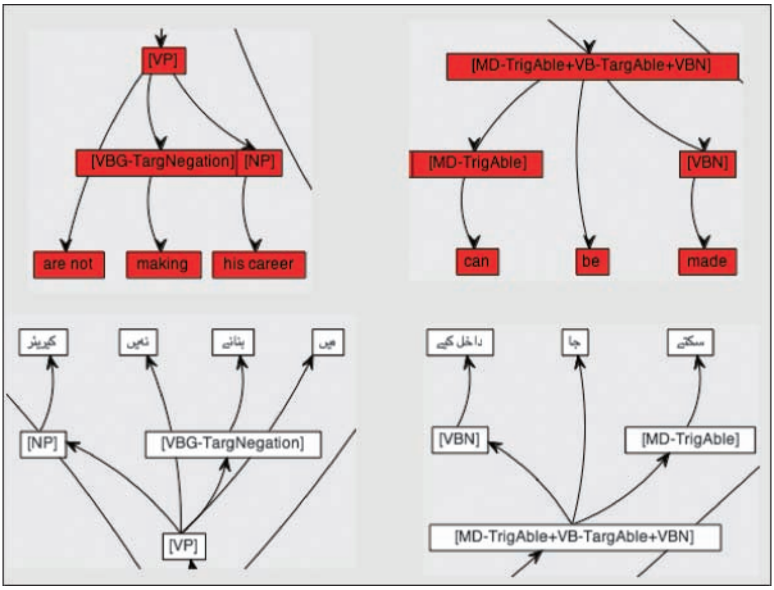


Figure 11
Example translation rules with tags for modality, negation, and entities combined with syntactic categories.

Table 2
Modality tags with their negated versions. Note that *Require* and *Permit* are in a dual relation, and thus RequireNegation is represented as NOTPermit and PermitNegation is represented as NOTRequire.

Require	NOTRequire
Permit	NOTPermit
Succeed	NOTSucceed
SucceedNegation	NOTSucceedNegation
Effort	NOTEffort
EffortNegation	NOTEffortNegation
Intend	NOTIntend
IntendNegation	NOTIntendNegation
Able	NOTAble
AbleNegation	NOTAbleNegation
Want	NOTWant
WantNegation	NOTWantNegation
Belief	NOTBelief
BeliefNegation	NOTBeliefNegation
Firm_Belief	NOTFirm_Belief
Firm_BeliefNegation	NOTFirm_BeliefNegation
Negation	

but *I failed to win* is tagged as "SucceedNegation," and *I did not fail to win* is tagged as "NOTSucceedNegation."

The tree-grafting algorithm proceeds as follows. For each tagged sentence, we iterate over the list of semantic tags. For each semantic tag, there is an associated word or sequence of words. For example, the modality tag TargAble may tag the word *swim*.

For each semantically tagged word, we find the parent node in the corresponding syntactic parse tree that dominates that word. For a word sequence, we find and compare the parent nodes for all of the words. Each node in the syntax tree has a category label. The following tests are then made and tree grafts applied:

- If there is a single node in the parse tree that dominates all and only the words with the semantic tag, graft the name of the semantic tag onto the highest corresponding syntactic constituent in the tree. For example, in Figure 10, which shows the grafting process for modality tagging, the semantic tag TargNOTAble that "hand over" receives is grafted onto the VB node that dominates all and only the words "hand over." Then the semantic tag TargNOTAble is passed up the tree to the VP node, which is the highest corresponding syntactic constituent.
- If the semantic tag corresponds to words that are adjacent daughters in a syntactic constituent, but less than the full constituent, insert a node dominating those words into the parse tree, as a daughter of the original syntactic constituent. The name of the semantic tag is grafted onto the new node and becomes its category label. This is a case of tree augmentation by node insertion.
- If a syntactic constituent selected for grafting has already been labeled with a semantic tag, overlay the previous tag with the current tag. We chose to tag in this manner simply because our system was not set up to handle the grafting of multiple tags onto a single constituent. An example

of this occurs in the sentence “The Muslims had obtained Pakistan.” If the NP node dominating *Pakistan* is grafted with a named entity tag such as NP-GPE, we overlay this with the NP-TargSucceed tag in a modality tagging scheme.

- In the case of a word sequence, if the words covered by the semantic tag fall across two different syntactic constituents, do nothing. This is a case of crossing brackets.

Our tree-grafting procedure was simplified to accept a single semantic tag per syntactic tree node as the final result. The algorithm keeps the last tag seen as the tag of precedence. In practice, we established a precedence ordering for modality/negation tags over named entity tags by grafting named entity tags first and modality/negation second. Our intuition was that, in case of a tie, finer-grained verbal categories would be more helpful to parsing than finer-grained nominal categories.[2] In cases where a word was tagged both as a MN target and a MN trigger, we gave precedence to the target tag. This is because, although MN targets vary, MN triggers are generally identifiable with lexical items. Finally, we used the simplified specificity ordering of MN tags described in Section 5.2 to ensure precedence of more specific tags over more general ones. Table 2 lists the modality/negation types from highest (Require modality) to lowest (Negation) precedence.[3]

7.3 SIMT Results

We evaluated our tree grafting approach by performing a series of translation experiments. Each version of our translation system was trained on the same bilingual training data. The bilingual parallel corpus that we used was distributed as part of the 2008 NIST Open Machine Translation Evaluation Workshop.[4] The training set contained 88,108 Urdu–English sentence pairs, and a bilingual dictionary with 113,911 entries. For our development and test sets, we split the NIST MT-08 test set into two portions (with each document going into either test or dev, and preserving the genre split). Our test set contained 883 Urdu sentences, each with four translations into English, and our dev set contained 981 Urdu sentences, each with four reference translations. To extract a syntactically informed translation model, we parsed the English side of the training corpus using a Penn Treebank–trained parser (Miller et al. 1998). For the experiments that involved grafting named entities onto the parse trees, we tagged the English side of the training corpus with the Phoenix tagger (Richman and Schone 2008). We word-aligned the parallel corpus with the Berkeley aligner. All models used a 5-gram language model trained on the English Gigaword corpus (v5) using the SRILM toolkit with modified KN smoothing. The Hiero translation grammar was extracted using the Joshua toolkit (Li et al. 2009). The other translation grammars were extracted using the SAMT toolkit (Venugopal and Zollmann 2009).

[2] In testing we found that grafting named entities first and MN last yielded a slightly higher BLEU score than the reverse order.

[3] Future work could include exploring additional methods of resolving tag conflicts or combining tag types on single nodes, for example, by inserting multiple intermediate nodes (effectively using unary rewrite rules) or by stringing tag names together.

[4] `http://www.itl.nist.gov/iad/mig/tests/mt/2008/doc/`.

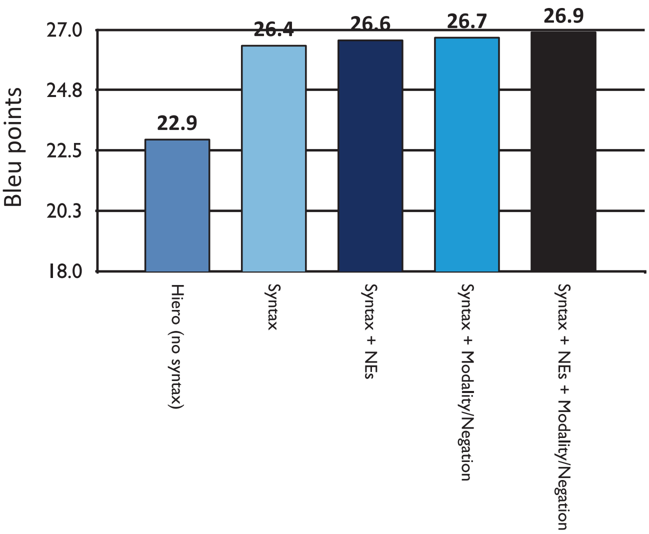


Figure 12
Results for a range of experiments conducted during the SIMT effort show the score for our top-performing baseline systems derived from a hierarchical phrase-based model (Hiero). Substantial improvements obtained when syntax was introduced along with feature functions (FFs) and further improvements resulted from the addition of semantic elements. The scores are lowercased BLEU calculated on the held-out devtest set. NE = named entities.

Figure 12 gives the results for a number of experiments conducted during the SIMT effort.[5] The experiments are broken into three groups: baselines, syntax, and semantics. To contextualize our results we experimented with a number of different baselines that were composed from two different approaches to statistical machine translation—phrase-based and hierarchical phrase-based SMT—along with different combinations of language model sizes and word aligners. Our best-performing baseline was a Hiero model. The Bleu score for this baseline on the development set was 22.9 Bleu points.

After experimenting with syntactically motivated grammar rules, we conducted experiments on the effects of incorporating semantic elements (e.g., named entities and modality/negation) into the translation grammars. In our devtest set our taggers tagged on average 3.5 named entities per sentence and 0.35 MN markers per sentence. These were included by grafting modality, negation, and named-entity markers onto the parse trees. Individually, each of these made modest improvements over the syntactically informed system alone. Grafting named entities onto the parse trees improved the Bleu score by 0.2 points. Modality/negation improved it by 0.3 points. Doing both simultaneously had an additive effect and resulted in a 0.5 Bleu score improvement over syntax alone. This improvement was the largest improvement that we got from anything other than the move from linguistically naive models to syntactically informed models.

We used bootstrap resampling to test whether the differences in Bleu scores were statistically significant (Koehn 2004). All of the results were a significant improvement over Hiero (at $p \leq 0.01$). The difference between the syntactic system and the syntactic system with named entities is not significant ($p = 0.38$). The differences between the

[5] These experiments were conducted on the devtest set, containing 883 Urdu sentences (21,623 Urdu words) and four reference translations per sentence. The BLEU score for these experiments is measured on uncased output.

syntactic system and the syntactic system with MN, and between the syntactic system and the syntactic system with both MN and named entities were both significant at $(p \leq 0.05)$.

Figure 13 shows example output from the final SIMT system in comparison to the pre-SIMT results and the translation produced by a human (reference). An error analysis of this example output illustrates that SIMT enhancements have resulted in the elimination of misleading translation output in several cases:

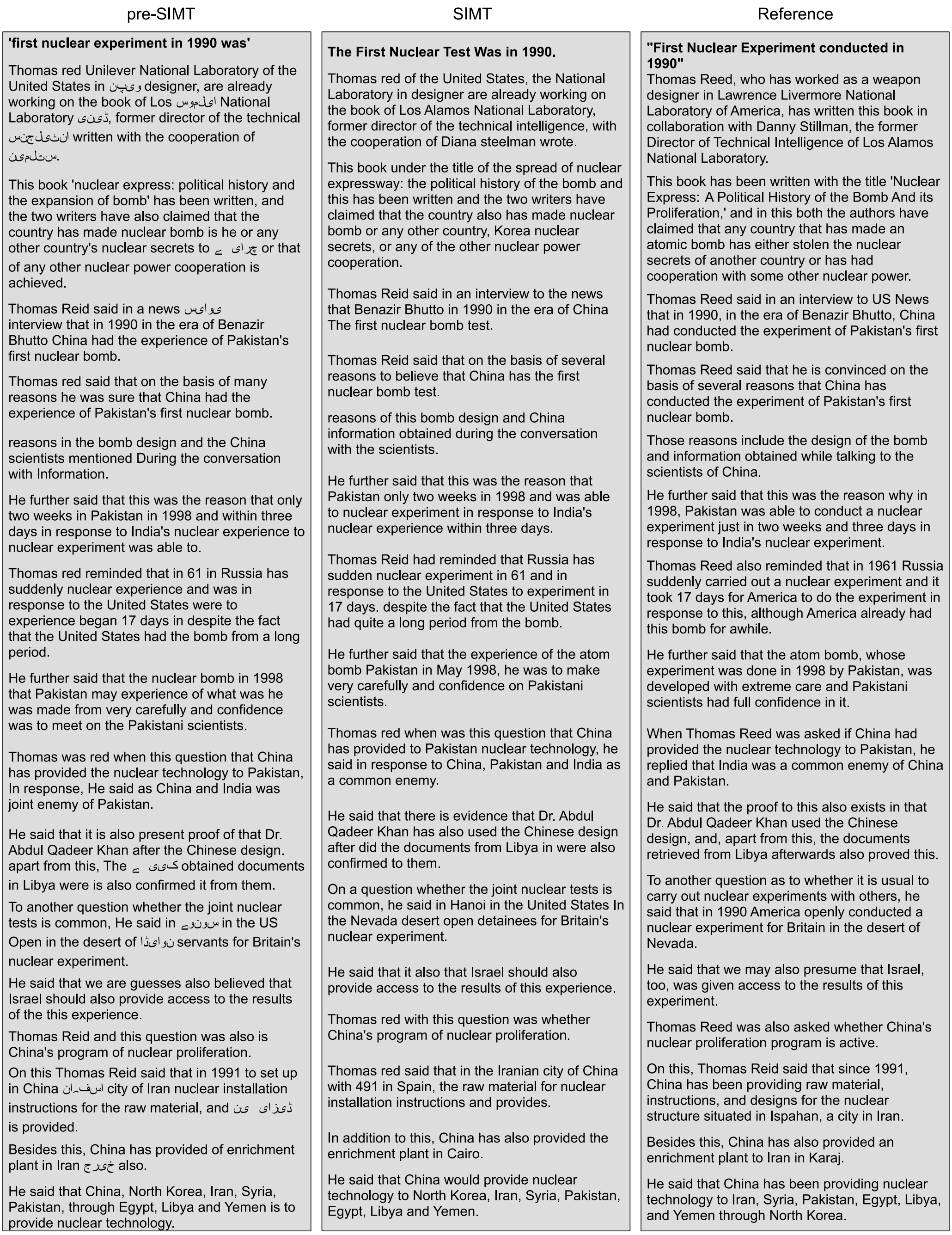
1. **pre-SIMT**: China had the experience of Pakistan's first nuclear bomb.
   SIMT: China has the first nuclear bomb test.
   reference: China has conducted the experiment of Pakistan's first nuclear bomb.
2. **pre-SIMT**: the nuclear bomb in 1998 that Pakistan may experience
   SIMT: the experience of the atom bomb Pakistan in May 1998
   reference: the atom bomb, whose experiment was done in 1998 by Pakistan
3. **pre-SIMT**: He said that it is also present proof of that Dr. Abdul Qadeer Khan after the Chinese design
   SIMT: He said that there is evidence that Dr. Abdul Qadeer Khan has also used the Chinese design
   reference: He said that the proof to this also exists in that Dr. Abdul Qadeer Khan used the Chinese design

The article in question pertains to claims by Thomas Reid that China allowed Pakistan to detonate a nuclear weapon at its test site. In the first example, however, the reader is potentially misled by the pre-SIMT output to believe that Pakistan launched a nuclear bomb on China. The SIMT output leaves out the mention of Pakistan, but correctly conveys the firm belief that the bomb event is a *test* (closely resembling the term *experiment* in the human reference), not a true bombing event. This is clearly an improvement over the misleading pre-SIMT output.

In the second example, the pre-SIMT output misleads the reader to believe that Pakistan is (or will be) attacked, through the use of the phrase *may experience*, where *may* is poorly placed. (We note here that this is a date translation error, i.e., the month of *May* should be next to the year 1998, further adding to the potential for confusion.) Unfortunately, the SIMT output also uses the term *experience* (rather than *experiment*, which is in the human reference), but in this case the month is correctly positioned in the output, thus eliminating the potential for confusion with respect to the modality. The lack of a modal appropriately neutralizes the statement so that it refers to an abstract event associated with the atom bomb, rather than an attack on the country.

In the third example, where the Chinese design used by Dr. Abdul Qandeer Khan is argued to be proof of the nuclear testing relationship between Pakistan and China, the first pre-SIMT output potentially leads the reader to believe that Dr. Abdul Qadeer is after the Chinese design (not that he actually used it), whereas the SIMT output conveys the firm belief that the Chinese design has been used by Dr. Abdul Qadeer. This output very closely matches the human reference.

Note that even in the title of the article, the SIMT system produces much more coherent English output than that of the linguistically naive system. The figure also shows improvements due to transliteration, which are described in Irvine et al. (2010). The scores reported in Figure 12 do not include transliteration improvements.



pre-SIMT	SIMT	Reference
'first nuclear experiment in 1990 was'	**The First Nuclear Test Was in 1990.**	**"First Nuclear Experiment conducted in 1990"**
Thomas red Unilever National Laboratory of the United States in ویپن designer, are already working on the book of Los ایلموس National Laboratory ڈینی, former director of the technical انٹیلجنس written with the cooperation of سٹلمین.	Thomas red of the United States, the National Laboratory in designer are already working on the book of Los Alamos National Laboratory, former director of the technical intelligence, with the cooperation of Diana steelman wrote.	Thomas Reed, who has worked as a weapon designer in Lawrence Livermore National Laboratory of America, has written this book in collaboration with Danny Stillman, the former Director of Technical Intelligence of Los Alamos National Laboratory.
This book 'nuclear express: political history and the expansion of bomb' has been written, and the two writers have also claimed that the country has made nuclear bomb is he or any other country's nuclear secrets to چرای ے or that of any other nuclear power cooperation is achieved.	This book under the title of the spread of nuclear expressway: the political history of the bomb and this has been written and the two writers have claimed that the country also has made nuclear bomb or any other country, Korea nuclear secrets, or any of the other nuclear power cooperation.	This book has been written with the title 'Nuclear Express: A Political History of the Bomb And its Proliferation,' and in this both the authors have claimed that any country that has made an atomic bomb has either stolen the nuclear secrets of another country or has had cooperation with some other nuclear power.
Thomas Reid said in a news یوایس interview that in 1990 in the era of Benazir Bhutto China had the experience of Pakistan's first nuclear bomb.	Thomas Reid said in an interview to the news that Benazir Bhutto in 1990 in the era of China The first nuclear bomb test.	Thomas Reed said in an interview to US News that in 1990, in the era of Benazir Bhutto, China had conducted the experiment of Pakistan's first nuclear bomb.
Thomas red said that on the basis of many reasons he was sure that China had the experience of Pakistan's first nuclear bomb.	Thomas Reid said that on the basis of several reasons to believe that China has the first nuclear bomb test.	Thomas Reed said that he is convinced on the basis of several reasons that China has conducted the experiment of Pakistan's first nuclear bomb.
reasons in the bomb design and the China scientists mentioned During the conversation with Information.	reasons of this bomb design and China information obtained during the conversation with the scientists.	Those reasons include the design of the bomb and information obtained while talking to the scientists of China.
He further said that this was the reason that only two weeks in Pakistan in 1998 and within three days in response to India's nuclear experience to nuclear experiment was able to.	He further said that this was the reason that Pakistan only two weeks in 1998 and was able to nuclear experiment in response to India's nuclear experience within three days.	He further said that this was the reason why in 1998, Pakistan was able to conduct a nuclear experiment just in two weeks and three days in response to India's nuclear experiment.
Thomas red reminded that in 61 in Russia has suddenly nuclear experience and was in response to the United States were to experience began 17 days in despite the fact that the United States had the bomb from a long period.	Thomas Reid had reminded that Russia has sudden nuclear experiment in 61 and in response to the United States to experiment in 17 days. despite the fact that the United States had quite a long period from the bomb.	Thomas Reed also reminded that in 1961 Russia suddenly carried out a nuclear experiment and it took 17 days for America to do the experiment in response to this, although America already had this bomb for awhile.
He further said that the nuclear bomb in 1998 that Pakistan may experience of what was he was made from very carefully and confidence was to meet on the Pakistani scientists.	He further said that the experience of the atom bomb Pakistan in May 1998, he was to make very carefully and confidence on Pakistani scientists.	He further said that the atom bomb, whose experiment was done in 1998 by Pakistan, was developed with extreme care and Pakistani scientists had full confidence in it.
Thomas was red when this question that China has provided the nuclear technology to Pakistan, In response, He said as China and India was joint enemy of Pakistan.	Thomas red when was this question that China has provided to Pakistan nuclear technology, he said in response to China, Pakistan and India as a common enemy.	When Thomas Reed was asked if China had provided the nuclear technology to Pakistan, he replied that India was a common enemy of China and Pakistan.
He said that it is also present proof of that Dr. Abdul Qadeer Khan after the Chinese design. apart from this, The کیی ے obtained documents in Libya were is also confirmed it from them.	He said that there is evidence that Dr. Abdul Qadeer Khan has also used the Chinese design after did the documents from Libya in were also confirmed to them.	He said that the proof to this also exists in that Dr. Abdul Qadeer Khan used the Chinese design, and, apart from this, the documents retrieved from Libya afterwards also proved this.
To another question whether the joint nuclear tests is common, He said in سونوے in the US Open in the desert of نوایڈا servants for Britain's nuclear experiment.	On a question whether the joint nuclear tests is common, he said in Hanoi in the United States In the Nevada desert open detainees for Britain's nuclear experiment.	To another question as to whether it is usual to carry out nuclear experiments with others, he said that in 1990 America openly conducted a nuclear experiment for Britain in the desert of Nevada.
He said that we are guesses also believed that Israel should also provide access to the results of the this experience.	He said that it also that Israel should also provide access to the results of this experience.	He said that we may also presume that Israel, too, was given access to the results of this experiment.
Thomas Reid and this question was also is China's program of nuclear proliferation.	Thomas red with this question was whether China's program of nuclear proliferation.	Thomas Reed was also asked whether China's nuclear proliferation program is active.
On this Thomas Reid said that in 1991 to set up in China اسفہان city of Iran nuclear installation instructions for the raw material, and ڈیزای ین is provided.	Thomas red said that in the Iranian city of China with 491 in Spain, the raw material for nuclear installation instructions and provides.	On this, Thomas Reid said that since 1991, China has been providing raw material, instructions, and designs for the nuclear structure situated in Ispahan, a city in Iran.
Besides this, China has provided of enrichment plant in Iran خیرج also.	In addition to this, China has also provided the enrichment plant in Cairo.	Besides this, China has also provided an enrichment plant to Iran in Karaj.
He said that China, North Korea, Iran, Syria, Pakistan, through Egypt, Libya and Yemen is to provide nuclear technology.	He said that China would provide nuclear technology to North Korea, Iran, Syria, Pakistan, Egypt, Libya and Yemen.	He said that China has been providing nuclear technology to Iran, Syria, Pakistan, Egypt, Libya, and Yemen through North Korea.

Figure 13
An example of the improvements to Urdu–English translation before and after the SIMT effort. Output is from the baseline Hiero model, which does not use linguistic information, and from the final model, which incorporates syntactic and semantic information.

8. Conclusions and Future Work

We developed a modality/negation lexicon and a set of automatic MN taggers, one of which—the structure-based tagger—results in 86% precision for tagging of a standard LDC data set. The MN tagger has been used to improve machine translation output by imposing semantic constraints on possible translations in the face of sparse training data. The tagger is also an important component of a language-understanding module for a related project.

We have described a technique for translation that shows particular promise for low-resource languages. We have integrated linguistic knowledge into statistical machine translation in a unified and coherent framework. We demonstrated that augmenting hierarchical phrase-based translation rules with semantic labels (through "grafting") resulted in a 0.5 Bleu score improvement over syntax alone.

Although our largest gains were from syntactic enrichments to the Hiero model, demonstrating success on the integration of semantic aspects of language bodes well for additional improvements based on the incorporation of other semantic aspects. For example, we hypothesize that incorporating relations and temporal knowledge into the translation rules would further improve translation quality. The syntactic grafting framework is well-suited to support the exploration of the impact of many different types of semantics on MT quality, though in this article we focused on exploring the impact of modality and negation.

An important future study is one that focuses on demonstrating whether further improvements in modality/negation identification are likely to lead to further gains in translation performance. Such a study would benefit from the inclusion of a more detailed manual evaluation to determine if modality and negation is adequately conveyed in the downstream translations. This work would be additionally enhanced through experimentation on other language pair(s) and larger corpora.

The work presented here represents the first small steps toward a full integration of MT and semantics. Efforts underway in DARPA's GALE program demonstrated the potential for combining MT and semantics (termed *distillation*) to answer the information needs of monolingual speakers using multilingual sources. Proper recognition of modalities and negation is crucial for handling those information needs effectively. In previous work, however, semantic processing proceeded largely independently of the MT system, operating only on the translated output. Our approach is significantly different in that it combines syntax, semantics, and MT into a single model, offering the potential advantages of joint modeling and joint decision-making. It would be interesting to explore whether the integration of MT with syntax and semantics can be extended to provide a single-model solution for tasks such as cross-language information extraction and question answering, and to evaluate our integrated approach (e.g., using GALE distillation metrics).

Acknowledgments

We thank Aaron Phillips for help with conversion of the output of the entity tagger for ingest by the tree-grafting program. We thank Anni Irvine and David Zajic for their help with experiments on an alternative Urdu modality/negation tagger based on projection and training an HMM-based tagger derived from Identifinder (Bikel, Schwartz, and Weischedel 1999). For their helpful ideas and suggestions during the development of the modality framework, we are indebted to Mona Diab, Eduard Hovy, Marge McShane, Teruko Mitamura, Sergei Nirenburg, Boyan Onyshkevych, and Owen Rambow. We also thank Basis Technology Corporation for their generous contribution of software components to this work. This work was supported, in part, by the Johns Hopkins Human Language

Technology Center of Excellence (HLTCOE), by the National Science Foundation under grant IIS-0713448, and by BBN Technologies under GALE DARPA/IPTO contract no. HR0011-06-C-0022. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsor.

References

Baker, Collin F., Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL '98, pages 86–90, Stroudsburg, PA.

Baker, Kathryn, Steven Bethard, Michael Bloodgood, Ralf Brown, Chris Callison-Burch, Glen Coppersmith, Bonnie J. Dorr, Nathaniel W. Filardo, Kendall Giles, Ann Irvine, Michael Kayser, Lori Levin, Justin Martineau, James Mayfield, Scott Miller, Aaron Phillips, Andrew Philpot, Christine Piatko, Lane Schwartz, and David Zajic. 2010a. Semantically informed machine translation. Technical Report 002, Human Language Technology Center of Excellence, Johns Hopkins University, Baltimore, MD.

Baker, Kathryn, Michael Bloodgood, Chris Callison-Burch, Bonnie J. Dorr, Nathaniel W. Filardo, Lori Levin, Scott Miller, and Christine Piatko. 2010b. Semantically-informed machine translation: A tree-grafting approach. In *Proceedings of The Ninth Biennial Conference of the Association for Machine Translation in the Americas*, Denver, CO.

Baker, Kathryn, Michael Bloodgood, Mona Diab, Bonnie J. Dorr, Ed Hovy, Lori Levin, Marjorie McShane, Teruko Mitamura, Sergei Nirenburg, Christine Piatko, Owen Rambow, and Gramm Richardson. 2010c. SIMT SCALE 2009—Modality annotation guidelines. Technical Report 004, Human Language Technology Center of Excellence, Johns Hopkins University, Baltimore, MD.

Baker, Kathryn, Michael Bloodgood, Bonnie J. Dorr, Nathanial W. Filardo, Lori Levin, and Christine Piatko. 2010d. A modality lexicon and its use in automatic tagging. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC)*, pages 1402–1407, Mediterranean Conference Center, Valletta.

Bar-Haim, Roy, Ido Dagan, Iddo Greental, and Eyal Shnarch. 2007. Semantic inference at the lexical-syntactic level. In *Proceedings of the 22nd National Conference on Artificial intelligence - Volume 1*, pages 871–876, Vancouver, British Columbia.

Bikel, Daniel M., Richard Schwartz, and Ralph M. Weischedel. 1999. An algorithm that learns what's in a name. *Machine Learning*, 34(1–3):211–231.

Böhmová, Alena, Silvie Cinková, and Eva Hajičová. 2005. A manual for tectogrammatical layer annotation of the Prague Dependency Treebank [English translation]. Technical Report #30, ÚFAL MFF UK, Prague, Czech Republic.

Chiang, David. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL-2005)*, pages 263–270, Ann Arbor, MI.

Diab, Mona T., Lori Levin, Teruko Mitamura, Owen Rambow, Vinodkumar Prabhakaran, and Weiwei Guo. 2009. Committed belief annotation and tagging. In *Proceedings of the Third Linguistic Annotation Workshop*, ACL-IJCNLP '09, pages 68–73, Stroudsburg, PA.

Farkas, Richárd, Veronika Vincze, György Móra, János Csirik, and György Szarvas. 2010. The CoNLL-2010 shared task: Learning to detect hedges and their scope in natural language text. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning—Shared Task*, CoNLL '10: Shared Task, pages 1–12, Stroudsburg, PA.

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

Hajič, Jan, Eva Hajičová, Petr Pajas, Jarmila Panevová, Petr Sgall, and Barbora Vidová Hladká. 2001. Prague Dependency Treebank 1.0 (Final Production Label), UFAL MFF UK, Prague, Czech Republic.

Huang, Bryant and Kevin Knight. 2006. Relabeling syntax trees to improve syntax-based machine translation quality. In *HLT-NAACL*, New York.

Irvine, Ann, Mike Kayser, Zhifei Li, Wren Thornton, and Chris Callison-Burch. 2010. Integrating output from specialized modules in machine translation: Transliteration in Joshua. *Proceedings of the Human Language Technology* *and North American Chapter of the Association for Computational Linguistics*, pages 240–247. *The Prague Bulletin of Mathematical Linguistics*, 93:107–116.

Klein, Dan and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 423–430, Sapporo, Japan.

Koehn, Philipp. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP 2004*, pages 388–395, Barcelona.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the ACL-2007 Demo and Poster Sessions*, Prague, Czech Republic, pages 177–180.

Kratzer, Angelika. 1991. Modality. In Arnim von Stechow and Dieter, editors, *Semantics: An International Handbook of Contemporary Research*. De Gruyter, Berlin, pages 639–650.

Larreya, Paul. 2009. Towards a typology of modality in language. In Raphael Salkie, Pierre Busuttil, and Johan van der Auwera, editors, *Modality in English: Theory and Description*. Mouton de Gruyter, Paris, pages 9–30.

Li, Zhifei, Chris Callison-Burch, Chris Dyer, Sanjeev Khudanpur, Lane Schwartz, Wren Thornton, Jonathan Weese, and Omar Zaidan. 2009. Joshua: An open source toolkit for parsing-based machine translation. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 135–139, Athens.

Marcus, Mitchell P., Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

McShane, Marjorie, Sergei Nirenburg, and Ron Zacharski. 2004. Mood and modality: Out of the theory and into the fray. *Natural Language Engineering*, 19(1):57–89.

Miller, Scott, Heidi Fox, Lance Ramshaw, and Ralph Weischedel. 1998. SIFT: Statistically-derived information from text. In *Seventh Message Understanding Conference (MUC-7)*, Washington, DC,

Miller, Scott, Heidi J. Fox, Lance A. Ramshaw, and Ralph M. Weischedel. 2000. A novel use of statistical parsing to extract information from text. In *Proceedings of Applied Natural Language Processing and the North American Association for Computational Linguistics*, pages 226–233, Seattle, Washington.

Murata, Masaki, Kiyotaka Uchimoto, Qing Ma, Toshiyuki Kanamaru, and Hitoshi Isahara. 2005. Analysis of machine translation systems' errors in tense, aspect, and modality. In *Proceedings of the 19th Asia-Pacific Conference on Language, Information and Computing* (PACLIC 2005), Taipei, Taiwan.

Nairn, Rowan, Cleo Condorovdi, and Lauri Karttunen. 2006. Computing relative polarity for textual inference. In *Proceedings of the International Workshop on Inference in Computational Semantics* (ICoS-5), pages 66–76, Buxton, England.

Palmer, Martha, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31:71–106.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002)*, pages 311–318, Philadelphia, PA.

Petrov, Slav and Dan Klein. 2007. Learning and inference for hierarchically split PCFGs. In *Proceedings of the 22nd American Association for Artificial Intelligence*, pages 1663–1666, Vancouver, British Columbia, Canada.

Prabhakaran, Vinodkumar, Owen Rambow, and Mona Diab. 2010. Automatic committed belief tagging. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 1014–1022, Beijing, China.

Prasad, Rashmi, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The Penn Discourse TreeBank 2.0. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*, pages 28–30, Marrakech.

Pustejovsky, James, Marc Verhagen, Roser Saurí, Jessica Littman, Robert Gaizauskas, Graham Katz, Inderjeet Mani, Robert Knippen, and Andrea Setzer. 2006. *TimeBank 1.2*. Linguistic Data Consortium, Philadelphia, PA.

Richman, Alexander and Patrick Schone. 2008. Mining wiki resources for multilingual named entity recognition.

In *Proceedings of ACL-08: HLT*, pages 1–9, Columbus, OH.

Rubin, Victoria L. 2007. Stating with certainty or stating with doubt: Intercoder reliability results for manual annotation of epistemically modalized statements. In *Proceedings of the Human Language Technology and North American Chapter of the Association for Computational Linguistics (Short Papers)*, pages 141–144, Rochester, NY.

Saurí, Roser and James Pustejovsky. 2009. FactBank: A corpus annotated with event factuality. *Language Resources and Evaluation*, 43(3):227–268.

Saurí, Roser, Marc Verhagen, and James Pustejovsky. 2006. Annotating and recognizing event modality in text. In *Proceedings of the 19th International Florida Artificial Intelligence Research Society Conference*, pages 333–339, Melbourne Beach, FL.

Sigurd, Bengt and Barbara Gawrónska. 1994. Modals as a problem for MT. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING) - Volume 1*, pages 120–124, Kyoto, Japan.

Steedman, Mark. 1999. Alternating quantifier scope in CCG. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL)*, College Park, MD.

Szarvas, György, Veronika Vincze, Richárd Farkas, and János Csirik. 2008. The BioScope corpus: Annotation for negation, uncertainty and their scope in biomedical texts. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing*, pages 38–45, Stroudsburg, PA.

van der Auwera, Johan and Andreas Ammann. 2005. Overlap between situational and epistemic modal marking. In Martin Haspelmath, Matthew S. Dryer, David Gil, and Bernard Comrie, editors, *World Atlas of Language Structures*. Oxford University Press, New York, chapter 76, pages 310–313.

Venugopal, Ashish and Andreas Zollmann. 2009. Grammar based statistical MT on Hadoop: An end-to-end toolkit for large scale PSCFG based MT. *Prague Bulletin of Mathematical Linguistics*, 91:67–78.

Venugopal, Ashish, Andreas Zollmann, and Stephan Vogel. 2007. An efficient two-pass approach to synchronous-CFG driven statistical MT. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL-2007)*, pages 500–507, Rochester, NY.

von Fintel, Kai and Sabine Iatridou. 2006. How to say *ought* in foreign: The composition of weak necessity modals. In *Proceedings of the 6th Workshop on Formal Linguistics*, Florianopolis, Brazil, August 2006.

Wang, Wei, Jonathan May, Kevin Knight, and Daniel Marcu. 2010. Re-structuring, re-labeling, and re-aligning for syntax-based machine translation. *Computational Linguistics*, 36(2):247–277.

Webber, Bonnie, Aravid Joshi, Matthew Stone, and Alistair Knott. 2003. Anaphora and discourse structure. *Computational Linguistics*, 29:545–587.

Wiebe, Janyce, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, 39(2–3):165–210.

Wilson, Theresa, Janyce Wiebe, and Paul Hoffman. 2009. Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational Linguistics*, 35:399–433.

Zollmann, Andreas and Ashish Venugopal. 2006. Syntax augmented machine translation via chart parsing. In *Proceedings on the Workshop on Statistical Machine Translation*, pages 138–141, New York City.