Rich Annotation Guided Learning

Xiang Li
Computer Science Department
Queens College, CUNY
New York, USA
jackieiuu729@gmail.com

Heng Ji
Computer Science Department
Queens College and Graduate Center, CUNY
New York, USA
hengjicuny@gmail.com

Faisal Farooq
Innovation Center
Siemens Medical Solutions
Malvern, USA
f.farooq@siemens.com

Hao Li
Computer Science Department
Graduate Center, CUNY
New York, USA
haoli.qc@gmail.com

Wen-Pin Lin
Computer Science Department
Queens College, CUNY
New York, USA
danniellin@gmail.com

Shipeng Yu
Innovation Center
Siemens Medical Solutions
Malvern, USA
shipeng.yu@siemens.com

***Abstract*—Supervised learning methods rely heavily on the quantity and quality of annotations provided by humans. As more natural language processing systems utilize human labeled data, it becomes beneficial to discover some hidden privileged knowledge from human annotators. In a traditional framework, a human annotator and a system are treated as isolated black-boxes. We propose better utilization of the valuable knowledge possessed by human annotators in the system development. This can be achieved by asking annotators to provide "rich annotations" for feature encoding. The rich annotations can come at multiple levels such as highlighting and generalizing contexts, and providing high-level comments. We propose a general framework to exploit such rich annotations from human annotators. This framework is a novel extension of our previous work by adding two more levels of rich annotations and two more systematic case studies. To demonstrate the power, generality and scalability of this approach, we apply the method in four very different applications in various domains: medical concept extraction, name translation, residence slot filling and event modality detection. Since richer annotations come at a higher cost (for example, take more time), we investigated the trade-off between system performance and annotation cost, when adding rich annotations from various levels. Experiments showed that the systems trained from rich annotations can save up to 65% annotation cost in order to obtain the same performance as using basic annotations. Our approach is able to bridge the gap between human annotators and systems in a seamless manner and achieve significant absolute improvement (6% - 15%) over state-of-the-art systems for all of these applications.**

***Keywords*-rich annotation; feature engineering**

I. Introduction

As an inter-disciplinary area, statistical natural language processing (NLP) requires two crucial aspects: (1) good choice of machine learning algorithms; (2) good feature engineering. In particular, (2) significantly affects the performance of systems. Linguistic annotation is a fundamental and crucial step of supervised learning. However, feature engineering remains a challenging task because it encompasses feature design, feature selection, feature induction and studies of feature impact, all of which are very time-consuming, especially when there are a lot of data or errors to analyze. As a result, in a typical feature engineering process, the system developer is only able to select a representative data set as the development set and analyze partial errors. Moreover, annotated corpora are usually prepared by a separate group of human annotators before system development. As a result, almost all of the previous NLP systems only utilized direct manual labels for training, while ignoring the valuable knowledge that human annotators have learned and summarized from corpora preparation. In fact, compared to system developers who normally design features based on partial data analysis, human annotators are usually more knowledgeable because they need to go through the entire data set and restrictively follow annotation guidelines.

To draw a parallel, if we consider an NLP system as a "*student*" while a human annotator as a "*teacher*", then the answers or grades (i.e., basic annotations) are just a small part of the pedagogy. Besides grading, a teacher also provides explanations and insights about why an answer is correct or incorrect, comments about what kind of further knowledge the student can benefit from, and how this can be further generalized. Similarly, besides using a text book, a teacher can also highlight part of the content or compose lecture notes. All of these additional evidences and comments can be considered as "*rich annotations*". When human annotators provide certain labels, they rely on certain rationales for the annotation of each instance. The feature engineering is expected to serve as a surrogate for this implicit knowledge. However, in order for that to be accomplished, large amounts of annotated data and highly specialized features are required which is often not feasible.

On the other hand, besides providing the basic annotations, the expert labellers can explicitly provide their rationales for those annotations, which can reduce the amount of training data and thus the annotation effort needed. The challenge then becomes two-fold:

1) feasibility of encoding this extra information such that the machine learning algorithms can exploit. Where as certain additional annotation can automatically be constructed into features, some others would require a systems developer to manually convert the additional information (such as comments) into features. Thus the burden on the system developer (specifically feature engineer) needs to be optimized such that the manual encoding (if necessary) does not require tremendous amount of effort or expertise.
2) the extra annotation effort involved needs to be acceptable to the annotators and the overall cost of the system. For example, simply highlighting the evidence in contexts would not add any significant burden where as generalizing enough knowledge to suggest what kind of linguistic features might be helpful adds slightly more effort. This calls for a fine compromise between the amount of additional information that can potentially make the system better and avoiding burden on the humans.

In this paper, we propose a new and general Rich Annotation Guided Learning (RAGL) framework in order to fill in the gap between an expert annotator and a feature engineer. As an extension of the comment-guided learning framework proposed in our previous work [1], this new framework aims to enrich features with the guidance of all levels of *rich annotations* from human annotators. We will also evaluate the comparative efficacy, generality and scalability of this framework by conducting case studies on four distinct applications in various domains: medical concept extraction, name translation, slot filling and event modality detection. Empirical studies demonstrate that with about little longer annotation time, we can significantly improve the performance for all tasks. We shall measure the annotation cost on these different domains so that this framework is also scalable. For example, the case study on event modality detection demonstrated that the system trained from rich annotations can save 65% annotation cost in order to obtain the same performance as using basic annotations.

The rest of this paper is structured as follows. Section II describes some related work. Section III presents an overview of our new learning framework incorporating rich annotations from human annotators. Section IV, Section V and Section VI present the detailed algorithms to incorporate rich annotations from various levels and four distinct case studies. Furthermore, Section IV illustrates the advantage of Level 1 on medical concept extraction, and Section V shows the contribution of Level 3 on two case studies, name translation and slot filling. After exploring both Level 1 and Level 3, Section VI applies all three levels to a single task, event modality detection, to compare the performance and investigate a trade-off provided by Level 2. Section VII then concludes the paper and sketches the possible future directions.

II. Related Work

In this section, we describe some related work about rich annotations and the applications.

A. Exploiting Rich Annotations

This paper is an extended version of our conference paper published at IMMM2011 [1]. In [1] we only asked human annotators to write down comments and suggestions that might improve re-scoring system output (Level 3 rich annotations) and provided two case studies. In this paper we extended rich annotations to Level 1, 2 and 3, and conducted systematic study on four case studies.

In some NLP tasks such as information retrieval, it's proven effective to incorporate user feedback to customize or tune a system, such as personalized search (e.g., [2]; [3]). However, such user feedback is not always available. Nevertheless most supervised learning methods rely on the labels by human annotators. Therefore there is great potential to fully utilize the deep knowledge from human annotators. [4] proposed to incorporate more of "*teacher's role*" (i.e., privileged knowledge) into traditional machine learning paradigm. We follow this basic idea and incorporate additional feedback from annotators into system development.

Several recent work has pointed out the problem that human annotators are "underutilized" and incorporated rich annotations into many classification problems [5], [6], [7]. Some other work [8], [9], [10] asked human annotators to label or select features. In this paper we shall generalize all kinds of annotator rationales into multiple levels and conduct a systematic study.

Castro et al. [11] investigated a series of human active learning experiments. Our experiment of using Rich Annotation Guided Learning to speed up human assessment exploited assistance from multiple systems.

Our idea of learning from error corrections is also similar to Transformation-based Error-Driven Learning, which has been successfully applied in many NLP tasks such as part-of-speech tagging [12], chunking [13], word sense disambiguation [14] and semantic role labeling [15]. In these applications the transformation rules are automatically learned based on sentence contexts at each iteration. However, our applications require global knowledge that may be derived from diverse linguistic levels and vary from one system to the other, and thus it's not straightforward to design and encode transformation templates. Therefore, in this paper

we choose a more modest way of exploiting the comments encoded by human annotators.

B. Applications

Information extraction from clinical text has recently received a lot of attention. Significant amount of this work in the literature has focused on areas such as radiology and pathology reports [16], [17]. For instance, Taira et al. [18], [19] have performed research on automatic structuring of radiology reports. More recently, researchers are making progress in the automated classification of clinical free text to code [17], [20] and applying machine learning and natural language processing for text mining in systems like BADGER [21], MedLEE [22] and CLEF [23]. Friedman et al [24] discussed the potential of using NLP techniques in the medical domain, and also provides a comparative overview of the state-of-the-art NLP tools applied to biomedical text. Literature in [24], [25], [26] provided a survey of various approaches to information extraction from biomedical text including named entity tagging and extracting relationship between different entities and between different texts. Of direct relevance is the analysis of doctors dictations by Chapman [26], which identified the seven most common uses of negation in doctors dictations. Some of the drawbacks of these works include: i) based on hard coded rules making them difficult to maintain and adapt [21], [23], ii) tuned for specific tasks (e.g., breast care reports [22] or pathology reports [27]) thus failing to generalize, iii) based on institution-specific styles, rules and guidelines (e.g., [28]). In all fairness, this is partially because high quality, labeled datasets of clinical documents have not been available. This is partly due to privacy laws and partly because they are expensive to create. Thus, learning valuable human (in this case clinical) knowledge during the course of annotation would significantly increase the quality of these systems and reduce the annotation efforts at the same time (given we posit that lesser data will need to be annotated).

Name translation is important well beyond the relative frequency of names in a text: a correctly translated passage, but with the wrong name, may lose most of its value. Most of the previous name translation work combined supervised transliteration approaches with Language Model based re-scoring ([29], [30]). Some recent research used parallel corpora or comparable corpora to re-score name transliterations ([31], [32]) or mine name translation pairs ([33], [34], [35], [36], [37], [38], [39], [40], [41]). However, most of these approaches required large amount of seeds and suffered from information extraction errors, and relied on phonetic similarity, context co-occurrence and document similarity to re-score candidate name translation pairs. In contrast, our approach described in this paper does not require any machine translation or transliteration features. Some recent work explored unsupervised or weakly-supervised name translation mining from large-scale data ([41], [42]) and Infoboxes ([43], [44], [45], [46], [47]). For example, Bouma et al. [44] aligned attributes in Wikipedia Infoboxes based on cross-page links; Ji et al. (2009) described various approaches to automatically mine name translation pairs from aligned phrases (e.g., cross-lingual Wikipedia title links) or aligned sentences (bi-texts). Some other work mined name translations from mono-lingual documents that include foreign language texts. For example, Lin et al. [48] described a parenthesis translation mining method; You et al. [49] applied graph alignment algorithm to obtain name translation pairs based on co-occurrence statistics. But none of these approaches exploited the feedback from human annotators.

There are many other alternative automatic assessment approaches for slot filling. Besides the RTE-KBP validation [50] discussed in the paper, some slot filling systems also conducted filtering and cross-slot reasoning (e.g., [51], [52]) to improve results.

Not many methods were proposed to address the problem of event modality attribute. [53] exploited surface features such as part-of-speech tags to detect event modalities and then applied them to improve event coreference resolution. Recent work by [54] described a statistical model based on annotations from rules and crowdsourcing tools. In the meanwhile, a linguistic corpus called "FactBank" with event semantic attributes has been developed by [55].

III. Rich Annotation Guided Learning

In this section, we present the general framework of incorporating rich human annotations into the learning process.

In Table I, we aim to formalize the mapping of some essential elements in human learning and machine learning for NLP.

In regular annotation interface, a human annotator is only asked to provide the final labels (e.g., 0/F or 1/T in binary settings). We call this as the basic annotation in '**Level 0**'. We can see that among these elements, little study has been conducted on incorporating rich annotations from human annotators. In most cases it was not the obligation for the human annotators to write down their evidence or comments during annotation. In contrast, the human learning scenario involves more interactions. However, we can assume that any annotator is able to verify and comment on his/her judgement. We propose to unleash the powerful knowledge based on rich annotations from human annotators on various deeper levels:

- **Level 1**: Ask an annotator to verify a label by providing surface evidence (e.g., highlighting indicative contexts);
- **Level 2**: Ask an annotator to verify a label by providing deep evidence (e.g., generalizing indicative contexts);
- **Level 3**: Ask an annotator to provide comments about linguistic features or resources that might be helpful for system development.

Table I
SOME ELEMENTS IN HUMAN LEARNING AND MACHINE LEARNING FOR NLP

Human Learning	Machine Learning for NLP	Approaches
student	system	baseline NLP system
teacher/ teaching assistant	human annotator/ human assessor	
textbook/ homework answer key	training data with basic annotations	
graded homework		
lecture notes/ graded homework with comments	**training data with rich annotations**	**Our proposed approach**
erroneous homework set	negative samples/errors	transformation based learning
homework review against lecture	system output with background documents	recognizing textual entailment
group study	pooled system responses	voting, learning-to-rank

Entry-level annotators are capable enough to provide rich annotations from Level 1 and Level 2, but we do need some annotators who have some certain knowledge about the task for Level 3 annotations. Based on this intuition we propose a new Rich Annotation Guided Learning (RAGL) paradigm as shown in Figure 1.

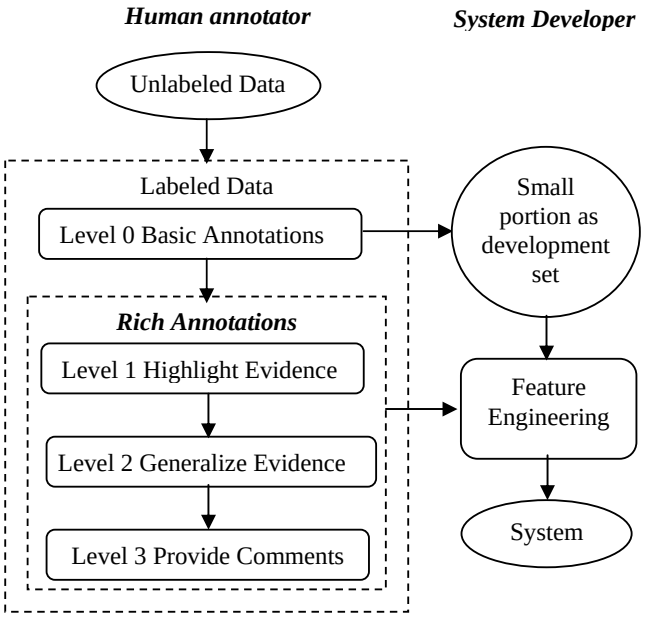

Figure 1. Rich Annotation Guided Learning Framework

IV. LEVEL 1: CHEAP RICH ANNOTATIONS

In this section, we introduce the framework to incorporate rich annotations from level 1, and then apply it to the case study on extracting medical concepts from clinical text.

A. General Framework

While the annotators are providing *basic annotations*, e.g., the class labels, we can often obtain richer annotations at almost no extra cost by highlighting part of text that leads the annotator to that conclusion (often called evidence or rationales). As described earlier in our discussion, this is the Level 1 of rich annotation. For instance, if the text contains the sentence *the patient has no history of alcohol abuse, and does not smoke*, the annotator will label it as *no* for the question whether the patient in question is a smoker or not. In addition, they can also highlight the evidence *does not smoke* since this part of the sentence leads to the no label. Providing this additional evidence adds marginally to the annotation effort, since the annotators would need to read the whole passage regardless, and highlighting the relevant part of the text would be simple if an easy-to-use graphical user interface is provided, such as selecting a contiguous piece of text using the mouse as in Figure 2.

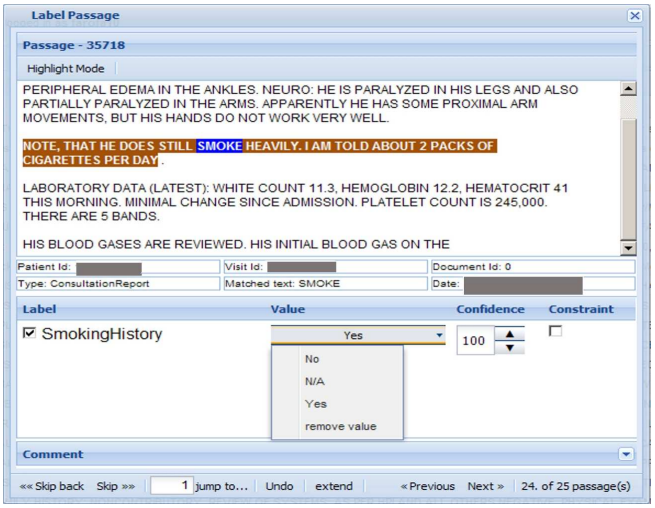

Figure 2. Providing Level 1 Rich Annotations

B. Case Study

For the purposes of this case study, we selected the problem of extracting medical concepts from clinical text. This problem lends itself directly to such a setting because

annotations (provided by clinical experts) are very expensive and often the systems not only have to yield the final answer (Y/N in binary cases), but also the evidence for that answer. Many of these information extraction tasks involve learning of certain medical concepts from the clinical free text, or learning to answer certain clinical questions about the patient. For instance, hospitals in the United States are required by CMS (Centers for Medicare & Medicaid Services) to submit answers of certain quality related questions (called quality measures) after patient discharge. In addition, as part of the meaningful use (MU) of Electronic Medical Records (EMR) initiative under the HITECH Act of the American Recovery and Reinvestment Act, certain key elements need to be reported as well. This is because actionable information that can be regularly and systematically mined from EMRs could lead to improved operational, financial, and clinical outcomes. The answers and the corresponding evidences are often found in the free text medical records (e.g., discharge summary) of the patient. Example questions include *Was the patient given aspirin within 24 hours of admission?*, *Did the adult patient smoke cigarettes anytime during the year prior to hospital arrival?* and *Was the patient assessed for rehabilitation services*. Since these concepts could be represented in different clinical terminologies e.g., rehabilitation assessment can be referred to as physical therapy, occupational therapy, PT, rehab etc., we concentrated on gathering information about five medical concepts with the help of expert medical personnel (such as expert chart abstractors). These concepts were chosen primarily due to their prevalence in quality reporting.

C. Experiments

In our implementation, we choose a state-of-the-art system [56] and show how using rich annotations can significantly improve the classification performance. With this annotation process, providing Level 1 rich annotations does not add a significant burden to the annotators, as they can perform this effortlessly while reading the text. In our experiments, we employed three clinical experts to serve as annotators. Based on past statistics, these annotators spent on average 1.5 hours to annotate 100 examples (of about 200 words each) without providing any rationale. In this new setting, we asked these experts to annotate new datasets with both the class labels as well as highlight rationales by selecting contiguous pieces of text. It was observed that with the rationale, annotation time changed marginally to 1.6 hours for 100 similar such examples, which is $\sim 10\%$ longer. In our experience, the additional annotation effort was acceptable.

Let x_i denote the features for an example text i computed from a dictionary of dimension d. Let $X = x_1, \ldots, x_N$ and $Y = y_1, \ldots, y_N$ denote the N training examples and their labels, respectively. In the training phase, we learn the model parameter $w \in \mathbb{R}^d$ by minimizing a cost function C between X and Y plus a regularization term on w, which can be denoted in the general form as

$$w^* = \arg\min_w \sum_{i=1}^{N} C(y_i, w^T x_i) + \lambda \cdot g(w)$$

with possible constraints on w. Here $g(w) \geqslant 0$ denotes the regularization term on w (such as $\|w\|^2$), and $\lambda \geqslant 0$ is the regularization parameter.

When the rich annotations are available, let $R = r_1, \ldots, r_N$ denote the highlighted evidences, where r_i denote the word sequence of highlighted evidence for example i. The objective is to learn the weight vector w such that the cost function C between X and Y, conditioned on the rich annotations R, is minimized (with regularization). Intuitively, the highlighted evidences provide additional insight to the assigned class label. Since each evidence r_i is simply a sequence of words, let us assume that additional features z_i of dimension s can be induced from the annotations for each example i. With this feature augmentation we can formulate the learning problem as

$$(w^*, v^*) = \arg\min_{w,v} \sum_{i=1}^{N} C(y_i, w^T x_i, v^T z_i) + \lambda_1 \cdot g(w) + \lambda_2 \cdot g(v)$$

with possible constraints on w and v, where $v \in \mathbb{R}^s$ is the weight vector for the evidence-induced feature z_i, and the regularization term involves both w and v (one can also assume a different regularization term for w and v). Then one can use the same solver as the standard binary classification to solve this optimization problem.

Table II
COMPARISON OF LEVEL 0 VS LEVEL 1 ANNOTATIONS FOR MEDICAL CONCEPT EXTRACTION

Medical Concepts	Level 0 AUC	Level 0 Accuracy	Level 1 AUC	Level 1 Accuracy
ST Elevation Assessed	0.87	0.82	0.96	0.90
Assessed for Rehabilitation	0.72	0.74	0.85	0.76
VTE Present on Arrival	0.90	0.83	0.95	0.89
Smoking History	0.72	0.61	0.83	0.80
Joint (e.g., knee) revision	0.87	0.75	0.94	0.80

Experiments were performed using actual EMR data from various medium/large-size hospitals. We built 5 datasets, one for each concept. The questions and the results for the two settings are shown in Table II. These passages were obtained from a set of ~ 10 million sentences by searching, in each case, for a few concepts of interest provided by the clinical experts. For each dataset, we first divided it into two subsets, one held out for testing only (30%) and one used for training (70%).

As the results show, there is significant improvement

across the board in all datasets, both for area under the ROC curve as well as absolute accuracies. This also means that same level of accuracies as Level 0 could have been achieved in the Level 1 setting by annotating fewer data, which would well compensate for the additional effort spent in highlighting.

D. Discussions

Level 1 approach assumes that the annotation evidence exists in the surface texts of the input data, and thus can represent them by simply highlighting such texts at the same time as producing labels. This hypothesis is not valid when a complicated task requires deep understanding of the contexts and external background knowledge. For comparison, we shall present a systematic study on incorporating Level 3 annotations in next section.

V. Level 3: Expensive Rich Annotations

In this section, we present the algorithm to incorporate rich annotations from level 3, and then apply it to the case studies on both name translation mining and slot filling.

A. Algorithm Overview

Recently many NLP tasks have moved from processing hundreds of documents to large-scale or even web-scale data. Once the collection grows beyond a certain size, it is not feasible to prepare a comprehensive answer key in advance. Because of the difficulty in finding information from a large corpus, any manually-prepared key is likely to be quite incomplete. Instead, we can pool the responses from various systems and have human annotators manually review and judge the responses. Assessing pooled system responses as opposed to identifying correct answers from scratch has provided a promising way to generate training data for NLP systems. Usually such tasks require deep knowledge beyond surface information provided by Level 0 and 1. In contrast, the comments from Level 3 can be exploited as features for automatic assessment. Then these features are manually constructed from the comments.

This algorithm aims to extensively incorporate all comments from an old development data set (i.e., "*old homework*" in human learning) into an automatic correction component. This assessor can be applied to improve the results for a new test data set (i.e., "*new homework*" in human learning).

The detailed algorithm can be summarized as follows.

1. The pipeline starts from running the baseline system to generate results. In this step we can also add the outputs from other systems (i.e., classmates in human learning) or even human annotators (i.e., Teaching Assistant (TA) in human learning). We will present one case study on slot filling that incorporates these two additional elements, and the other case study on name translation that only utilizes results from the baseline system.

2. We obtain comments from human annotators on a small development set D^i. Each time we ask a human annotator to pick up N (N=3 in this paper, the value of 3 was arbitrarily chosen; variations in this number of clusters produce only small changes in performance) random results to generate one new comment. One could impose some predefined format or template restrictions for the comments, such as marking the indicative words as rich annotations and encoding them as features. Nonetheless, we found that most of the expert comments are rather implicit and even requires global knowledge. Nonetheless these comments represent general solutions to reduce the common errors from the baseline system.

3. We encode these comments into features through manual construction. We then further train a Maximum Entropy (MaxEnt) based automatic assessor A^i using these features. For each response generated from the baseline system, A^i can classify it as correct or incorrect. We choose a statistical model instead of rules because heuristic rules may overfit a small sample set and highly dependent on the order. In contrast, MaxEnt model has the power of incorporating all comments into a uniform model by assigning weights automatically. In this way we can integrate assessment results tightly with comments during MaxEnt model training.

4. Finally, A^i is applied as a post-processing step to any new data set D^{i+1}, and filter out those results judged as incorrect.

The algorithm can be conducted in an iterative fashion. For example, human annotators can continue to judge and provide comments for D^{i+1} and we can update the automatic assessor to A^{i+1} and apply it to a new data set D^{i+2}, and so on.

We conduct case studies on two distinct application domains: a relatively simple name translation task(V-B) and a more challenging residence slot filling task(V-C).

B. Name Translation Mining

This section presents the first case study of applying Level 3 (human comments) guided learning for name translation validation.

1) Task Definition: Previous name translation pair mining approaches suffer from low accuracy and thus it is important to develop automatic methods to evaluate whether the mined name pairs are correct or not. For example, we need to determine whether the English name "*Michael Jackson*" and the Chinese name "*Mai Ke Er . Jie Ke Xun*" are a correct translation pair. In this paper we focus on validating person name translations by encoding the comments that human annotators made on a small data set.

2) Baseline System: We applied a simple weakly-supervised approach similar to [47] to mine name translation pairs from English and Chinese Wikipedia Infoboxes. A standard Wikipedia entry includes a title, a document describing the entry, and an "infobox", which is a fixed-format

table designed to be added to the top right-hand corner of the article to consistently present a summary of some unifying attributes about the entry. Based on the fact that some certain types of expressions are written in language-independent forms (such as dates and numbers), we generate seed name pairs automatically based on some simple facts (e.g., if two person entries had the same "*birth-date*" and "*death-date*" Infobox slot values, they are considered as a seed pair). Starting from these seeds, we then apply a bootstrapping algorithm based on Infobox slot comparison to mine more pairs iteratively. For example, after we get the seed translation pair of "*Mai Ke Er . Jie Ke Xun (Michael Jackson)*", we can iteratively get new pairs with a large portion of overlapped slots. For example, since "*Ji Xun Wu Ren Zu*" and "*The Jackson 5*" share many slot values such as "*member = Michael Jackson*" and "*years active = 1964-1990*", they are likely to be a translation pair. Next we can use the new pair of "*Ji Xun Wu Ren Zu (The Jackson 5)*" to mine more pairs such as "*Gang Cheng Chang Pian (Steeltown Records)*" their "*labels*".

3) Comments and Feature Encoding: The detailed comments used for validating name translations are as follows.

- **Comment 1: "*these two names do not co-occur often*"** This comment indicates that we can exploit global statistics to filter out some obvious errors, such as "*Ethel Portnoy*" and "*Chen Yao Zu*". Using Yahoo! search engine, we compute the co-occurrence, conditional probability and mutual information of a Chinese Name *CHName* and an English name *ENName* appearing in the same document from web-scale data with setting a threshold for each criteria.
- **Comment 2: "*these two names have very different pronunciations*"** Many foreign names are transliterated from their origin pronunciations. As a result, person name pairs (e.g., "*Lomana LuaLua*" and "*Luo Ma Na . Lu A Lu A*") usually share similar pronunciations. In order to address this comment, we define an additional feature based on the Damerau-Levenshtein edit distance ([57]; [58]) between the Pinyin form of *CHName* and *ENName*. Using this feature we can filter out many incorrect pairs, such as "*Maurice Dupras*" and "*Zhuo Ya . Ke Si Mo Jie Mi Yang Si Qia Ya*".
- **Comment 3: "*these two names have different profiles*"** When human annotators evaluate the name translation pairs, they often exploit their world knowledge. For example, they can quickly judge "*Comerford Walker*" is not a correct translation for "*Sen Gang Er Lang (Jiro Oka Mori)*" because they have different nationalities (one is U.S. while the other is Japan). To address this comment, we define the *profile* of a name as a list of attributes. Besides using all of the Wikipedia Infobox values, we also run a bi-lingual information extraction (IE) system [59] on large comparable corpora (English and Chinese Gigaword corpora) to gather more attributes for *ENName* and *CHName*. For example, since "*Nick Grinde*" is a "*film director*" while "*Yi Wan . Si Te Lan Si Ji*" is a "*physicist*" in these large contexts, we can filter out this incorrect name pair.

The detailed features converted from the above comments are summarized in Table III.

Table III
VALIDATION FEATURES FOR NAME TRANSLATION

Comments	Features
1	co-occurrence, conditional probability and mutual information of *CHName* and *ENName* appearing in the same document from web-scale data
	conditional probability of *CHName* and *ENName* appearing in the same document from web-scale data
	mutual information of *CHName* and *ENName* appearing in the same document from web-scale data
2	Damerau-Levenshtein edit distance between the Pinyin form of *CHName* and *ENName*
3	overlap rate between the attributes of *CHName* and the attributes of *ENName* according to Wikipedia Infoboxes
	overlap rate between the attributes of *CHName* and the attributes of *ENName* according to IE results of large comparable corpora

4) Data and Scoring Metric: We used English and Chinese Wikipedias as of November 2010, including 10,355,225 English pages and 772,826 Chinese pages, and mined 5368 name pairs, where 3719 pairs are correct pairs used as positive samples, and the rest 1649 pairs are incorrect pairs as negative samples. This also shows the capacity of rich annotations to target the problem of unbalanced data. A small set of 100 pairs is taken out as the development set for the human annotator to encode comments. The comment guided assessor is then trained and tested on the remaining pairs by 5-folder cross-validation.

It is time consuming to evaluate the mined name pairs because sometimes the human annotator needs Web access to check the contexts of the pairs, especially when the translations are based on meanings instead of pronunciations. We implemented a baseline of mining name pairs from cross-lingual titles in Wikipedia as an incomplete answer key, and so we only need to ask two human annotators (not system developers) to do manual evaluation on our system generated pairs, which are not in this answer key (1672 in total). A

name pair is judged as correct if both of them are correctly extracted and one is the correct translation of the other. Such a semi-automatic method can speed up evaluation. On average each human annotator spent about 3 hours on evaluation.

5) Overall Performance: Table IV shows Precision (P), Recall (R) and F-measure (F) scores before and after applying the comment guided assessor on name translation pairs. As we can see from Table IV, our approach achieved 28.7% absolute improvement on precision with a small loss (4.9%) in recall. In order to check how robust our approach is, we conducted the Wilcoxon Matched-Pairs Signed-Ranks Test on F-measures. The results show that we can reject the hypothesis that the improvements using Level 3 annotations were random at a 99.8% confidence level.

Table IV
THE IMPACT OF LEVEL 3 ANNOTATIONS ON NAME TRANSLATION

Annotation Type		P (%)	R (%)	F (%)
Basic Annotation	Level 0	69.3	100.0	81.9
Rich Annotation	Level 3	98.0	95.1	**96.5**

C. Slot Filling

In this section, we shall apply Level 3 annotations to a more challenging task of slot filling and investigate the detailed aspects of human comments guided learning by comparing it with other alternative methods.

1) Task Definition: In the slot filling task [60], [61], attributes (or "*slots*") derived from Wikipedia infoboxes are used to create the initial (or reference) knowledge base (KB). A large collection of source news and web documents is then provided to the slot filling systems to expand the KB automatically.

The goal of slot filling is to collect information regarding certain attributes of a query from the corpus. The system must determine from this large corpus the values of specified attributes of the entity. Along with each slot answer, the system must provide the ID of a document that supports the correctness of this answer.

We choose three residence slots for person entities ("*countries_of_residence*", "*stateorprovinces_of_residence*" and "*cities_of_residence*") for our case study because they are one group of the most challenging slot types, for which almost all systems perform poorly (less than 20% F-measure). For example, we need to decide whether it is true that the query "*Adam Senn*" has lived in the country "*America*" or in the city "*Paris*".

2) Baseline Systems: We use a slot filling system [51] that achieved highly competitive results (ranked at top 3 among 31 submissions from 15 teams) at the KBP2010 evaluation as our baseline. This system includes multiple pipelines in two categories: two bottom-up IE based approaches (pattern matching and supervised classification) and a top-down Question Answering (QA) based approach that search for answers constructed from target entities and slot types. The overall system begins with an initial query processing stage where query expansion techniques are used to improve recall. The best answer candidate sets are generated from each of the individual pipelines and are combined in a statistical re-ranker. The resulting answer set, along with confidence values are then processed by a cross-slot reasoning step based on Markov Logic Networks [62], resulting in the final system outputs. In addition, the system also exploited external knowledge bases such as Freebase [63] and Wikipedia text mining for answer validation.

In order to check how robust the RAGL assessor is, we also run it on some other anonymous systems in KBP2010 with representative performance (high, medium and low).

3) Comments and Feature Encoding: The detailed comments used for our slot filling experiment are as follows.

- **Comment 1: "*this answer is not a geo-political name*"**
  This comment is intended to address some obvious errors that could not be Geo-Political (GPE) names in any contexts. In order to address this comment, we apply a very large gazetteer of GPE hierarchy (countries, states and cities) from the geonames website (http://www.geonames.org/statistics/) for answer validation.
- **Comment 2: "*this answer is not supported by this document*"**
  Some answers obtained from Freebase may be incorrect because they are not supported by the source document. Answer validation was mostly conducted on the document basis, but for the residence slots we need to use sentence-level validation. In addition, some sentence segmentation errors occur in web documents. To address this comment, we apply a coreference resolution system [59] to the source document, and check whether any mention of the query entity and any mention of the candidate answer entity appear in the same sentence.
- **Comment 3: "*this answer is not a geo-political name in this sentence*"**
  Some ambiguous answers are not GPE names in certain contexts, such as "*European Union*". To address this comment, we extract the context sentences including the query and answer mentions, and run a name tagger [64] to verify the candidate answer is a GPE name.
- **Comment 4: "*this answer conflicts with this system/other system's output*"**
  When an answer from our system is not consistent with another answer that appears often in the pooled system responses, this comment suggests us to remove our answer. In order to address this comment, we implemented a feature based on hierarchical spatial reasoning. We conduct majority voting on all the available system responses, and collect the answers with global confidence values (voting

weights) into a separate answer set *ha*. Then for any candidate answer *a*, we check the consistency between *a* and any member of *ha* by name coreference resolution and part-whole relation detection based on the gazetteer of GPE hierarchy as described in Comment 1. For example, if "*U.S.*" appears often in *ha* we can infer "*Paris*" is unlikely to be a correct answer for the same query; on the other hand if "*New York*" appears often in *ha* we can confirm "*U.S.*" as a correct answer.

The detailed features converted from the above comments are summarized in Table V.

Table V
VALIDATION FEATURES FOR SLOT FILLING

Comments	Features
1	whether the answer is in the geo-political gazetteer
2	whether any mention of the query entity and any mention of the answer entity appear in the same sentence using coreference resolution
3	whether the answer is a GPE name by running name tagging on the context sentence
4	whether the answer conflicts with the other answers which received high votes across systems using inferences through the GPE hierarchy

4) Data and Scoring Metric: During KBP2010, an initial answer key annotation was created by Linguistic Data Consortium (LDC) through a manual search of the corpus, and then an independent adjudication pass was applied by LDC human annotators to assess these annotations together with pooled system responses to form the final gold-standard answer key. We incorporated the assessment comments for our system output on a separate development set (182 unique non-NIL answers in total) from KBP2010 training data set to train the automatic assessor. Then we conduct blind test on the KBP2010 evaluation data set that includes 1.7 million newswire and web documents. The testing data from our KBP system output consists of 25 positive samples and 121 negative samples, which is also unbalanced. The final answer key for the blind test set includes 81 unique non-NIL answers for 49 queries.

The number of features we can exploit is limited by the unknown restrictions of individual systems. For example, some other systems used distant learning based answer validation and so could not provide specific context sentences. Since comment 2 and comment 3 require context sentences, we trained one assessor using all features and tested it on our own system. Then, we trained another assessor using only comment 1 and 4 and tested it on three other systems representing different levels of performance.

Equivalent answers (such as "*the United States*" and "*USA*") are grouped into equivalence classes. Each system answer is rated as correct, wrong, or redundant (an answer that is equivalent to another answer for the same slot or an entry already in the knowledge base). Given these judgments, we calculate the precision, recall and F-measure of each system, as defined in [60], [61].

5) Overall Performance: Table VI shows the slot filling scores before and after applying the RAGL assessors (because of the KBP Track requirements and policies, we could not mention the specific names of other systems). The Wilcoxon Matched-Pairs Signed-Ranks Test show we can reject the hypothesis that the improvements using RAGL over our system were random at a 99.8% confidence level. It also indicates that the features encoded from comment 2 and comment 3 that require intermediate results such as context sentences helped boost the performance about 3.4%. We can see that although the other high-performing system may have used very different algorithms and resources from ours, our assessor still provided significant gains. Our approach improved the precision on each system (more than 200% relative gains) with some loss in recall. Since most comments focused on improving precision, F-measure gains for moderate-performing and low-performing systems were limited by their recall scores. This is similar to the human learning scenario where students from the same grade can learn more from each other than from different grades. In addition, the errors removed by our approach were distributed equally in newswire (48.9%) and web data (51.1%), which indicates the comments from human annotators reached a good degree of generalization across genres.

Table VI
OVERALL PERFORMANCE OF SLOT FILLING

Slot Filling Systems		Annotation Category	P (%)	R (%)	F (%)
Our system		Level 0	17.1	30.9	22.0
		Level 3 (f1+f4)	26.2	27.2	**26.7**
		Level 3 (full)	38.5	24.7	**30.1**
Other systems	High-Performing	Level 0	13.7	29.6	18.8
		Level 3 (f1+f4)	40.9	22.2	**28.8**
	Moderate-Performing	Level 0	12.2	7.4	9.2
		Level 3 (f1+f4)	35.7	6.2	**10.5**
	Low-Performing	Level 0	6.7	3.7	4.8
		Level 3 (f1+f4)	50	3.7	**6.9**

6) Cost and Contribution of Each Comment: The comments from the RAGL assessor may reflect different aspects of the system. Therefore it will be interesting to investigate what types of comments are most useful and not costly. We did another experiment by applying one comment at a time into the assessor. Table VII shows the results along with the cost of generating and encoding each comment (i.e., knowledge transferring to its corresponding feature), which was carefully recorded by the human annotators.

Table VII
COST AND CONTRIBUTION OF EACH COMMENT

Annotations		Level 0	Level 3			
			f1	f2	f3	f4
Performance	P (%)	17.1	17.6	26.4	26.7	25.6
	R (%)	30.9	30.9	28.4	28.4	27.2
	F (%)	22.0	22.4	27.4	27.5	26.3
Cost	#samples reviewed	-	3	3	3	3
	providing comments (minutes)	-	3	3	3	3
	encoding comments (minutes)	-	30	240	60	30

Table VII indicates that every feature made contributions to precision improvement. Comment 1 (gazetteer-based filtering) only provided limited gains mainly because our own system already extensively used similar gazetteers for answer filtering. This reflects a drawback of our comment generation procedure - the assessor had no prior knowledge about the approaches used in the systems. Comment 2 (using coreference resolution to check sentence occurrence) took most time to encode but also provides significant improvement. Comment 4 (consistency checking against responses with high votes) provided significant gains in precision (8.5%) but also some loss in recall (3.7%). The problem was that systems tend to make similar mistakes, and the human annotator was biased by those correct answers that appeared frequently in the pooled system output. However, Comment 4 was able to filter out many errors that are otherwise very difficult to detect. For example, because "*Najaf*" appears very often as a "*cities_of_residence*" in the pooled system responses, Comment 4 successfully removed six incorrect "*countries_of_residence*" answers for the same query: "*Syrian*", "*Britain*", "*Iranian*", "*North Korea*", "*Saudi Arabia*" and "*United States*". On the other hand, Comment 4 confirmed correct answers such as "*New York*" from "*Brooklyn*", "*Texas*" from "*Dallas*", "*California*" and "*US*" from "*Los Angeles*".

7) Impact of Data Size: We also did a series of runs to examine how our own system performed with different amounts of training data. These experiments are summarized in Figure 3. It clearly shows that the learning curve converges quickly. Therefore, we only need a very small amount of training data (36 samples, 20% of total) in order to obtain similar gains (6.8%) as using the whole training set.

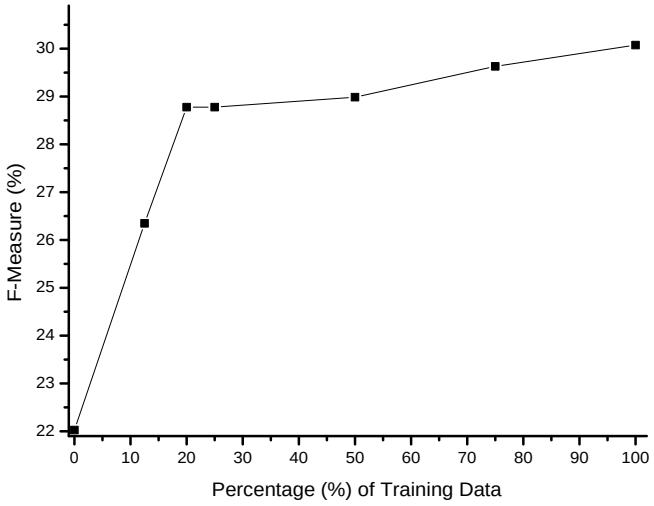

Figure 3. Impact of Training Data Size

8) Speed up Human Assessment: Human assessment for slot filling is also a costly task because it requires the annotators to judge each answer against the associated source document. Since our RAGL approach achieved positive impact on system output, can it be used to as feedback to speed up human assessment? We applied the RAGL assessor trained from comment 1 and comment 4 to the top 13 KBP systems for KBP2010 evaluation set. We automatically ranked the pooled system responses of residence slots according to their confidence values from high to low.

For comparison, we also exploited the following methods:

- **Baseline**
  As a baseline, we ranked the responses according to the alphabetical order of slot type, query ID, query name and answer string and doc ID. This is the same approach used by LDC human annotators for assessing KBP2010 system responses.
- **Oracle (Upper-Bound)**
  We used an oracle (for upper-bound analysis) by always assessing all correct answers first.

Figure 4 summarizes the results from the above 3 approaches. For this figure, we assume a labor cost for assessment proportional to the number of non-NIL items assessed. Note that all redundant answers are also included in these counts because human annotators also spent time on assessing them. This is only approximately correct; it

may be faster (per response) to assess more responses to the same slot. The common end point of curves represents the cost and benefit of assessing all system responses. We can see that if we employ the RAGL assessor and apply some cut-off, the process can be dramatically more efficient than the regular baseline based on alphabetical order. For example, in order to get 79 correct answers (76% of total), RAGL approach took human annotators only 5.5 hours, while the baseline approach took 13.4 hours.

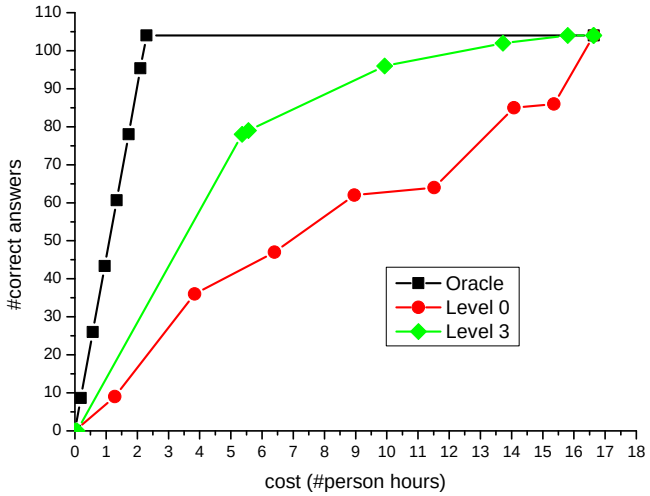

Figure 4. Human Assessment Method Comparison

9) Comparison with Alternative Methods: An alternative approach to validate answers is to use textual entailment techniques as in the RTE-KBP validation task [50], [65], which was partly inspired by CLEF Question Answering task [66]. This task consists of determining whether a candidate answer (hypothesis "*H*") is supported in the associated source document (text "*T*") using entailment techniques. For the residence slots, we are considering in this paper, they treat each context document as a "*T*", and apply pre-defined sentence templates such as "*[Query] lived in [Answer]*" to compose a "*H*" from system output. Entailment and reasoning methods from the TAC-RTE2010 systems are then applied to validate whether "*H*" is true or false according to "*T*". These RTE-KBP systems are limited to individual *H-T* instances and optimized only on a subset of the pooled system responses. As a result, they aggressively filtered many correct answers and did not provide improvement on most slot filling systems (including the representative ones we used for our experiment). In contrast, our RAGL approach has the advantage of exploiting the generalized knowledge and feedback from assessors across all queries and systems.

D. Discussions

We have demonstrated that the comments from Level 3 provided significant improvement for two distinct applications, which require deep understanding of the contexts beyond surface texts. However, we also observed that some comments still require a system developer to fully understand and transfer the knowledge into detailed feature encoding by incorporating external resources. Therefore, the additional cost may vary based on the clarity of each comment and the availability of linguistic resources. In next section we will focus on investigating whether Level 2 is a good trade-off approach between performance gains and cost.

VI. Level 2: A Trade-off

In this section, we will compare three levels of rich annotations by applying all of them to one single task of event modality detection, and investigate whether Level 2 rich annotations can provide a trade-off.

A. Task Definition

We conduct a case study on predicting *Modality* attributes for events defined by the Automatic Content Extraction (ACE) evaluations (http://nist.gov/speech/tests/ace). The Modality attribute indicates whether an event really took place. An event is "Asserted" when the author or speaker makes reference to it as though it were a real occurrence, such as "*A car bomb **exploded** Thursday in the heart of Jerusalem, **killing** at least two people, police said.*". All other events will be annotated as "OTHER", such as "*He asked the committee to **accept** his paper.*" (Commanded and Requested Events), and "*Promises of **aid** made by Arab and European countries.*" (Threatened, Proposed and Discussed Events). The annotators were trained to follow the ACE2005 event annotation guideline: http://projects.ldc.upenn.edu/ace/docs/English-Events-Guidelines_v5.4.3.pdf.

B. General Model

We use a MaxEnt based classifier to detect the modality attribute of a given event instance. This model has the power of assigning weights automatically to features from all levels of rich annotations. During the annotation process, annotators were asked to provide different levels of rich annotations for training data, and we then encoded such rich annotations into a MaxEnt model, as illustrated in the following subsections.

C. Level 0: Baseline

In Level 0, the annotators were asked to label each instance as "Asserted" or "Other" without providing additional markups or comments. During the baseline system development, we selected an n-gram *ng* (n=1, 2, 3) as an indicative context if it matches one of the following two

conditions: (1) *ng* appears only in "Other" events, and with frequency higher than a threshold δ. (2) (the frequency of *ng* occurring in "Other" events) / (the frequency of *ng* occurring in "Asserted" events) is higher than a threshold θ. Both δ and θ were optimized from a small development set including 30 events. The baseline feature is based on the number of indicative context n-grams. For example, given a "*Movement_Transport*" event in "*Bush and Putin were scheduled to leave straight after their talks for the Group of Eight summit of the largest industrialised nations in Evian, France.*" triggered by "*leave*", the indicative context n-grams are "*leave*", "*straight*" and "*to leave*", thus the feature value is 3.

D. Level 1: Highlight Contexts

In Level 1, we ask the human annotators to highlight indicative contexts while labeling modality attribute of each event. The features implemented by the system developers are based on the number of indicative context n-grams, which are highlighted by human annotators. Table VIII presents examples with "Other" modalities and their corresponding highlighted contexts for both Level 1 and Level 2.

E. Level 2: Generalize Contexts

The highlighted features from Level 1 are effective if the contexts are explicit. However, in many other cases the annotators may want to highlight categories of some certain evidence, to indicate informative semantic concepts, templates or patterns that are beyond bag-of-words. For example, instead of highlighting "*scheduled to*", the annotator may want to generalize a category of "*words indicating planning*" because other phrases such as "*plan for*" can play the same role. In Level 2, the human annotators marked up the categories of trigger words and contexts as shown in Table IX, which may indicate "Other" modality. In this Level, annotators only suggest category names, and system developers try to acquire contextual word clusters for each category.

Table IX
CONTEXTUAL CATEGORIES FOR LEVEL 2

Category Name	Size	Example Words/Phrases
Verb	--	(Check whether the event trigger word is a verb)
Modal Auxiliary	10	"could", "would" and "might"
Uncertainty	21	"perhaps", "maybe", "possibly", and "likely"
Planning	11	"planned" and "scheduled"
Assumption	72	"supposed", "estimated" and "expected"
Negation	136	"barely", "impossible", "never", and "declined"

An added advantage of this level of richer annotation is the ease of translation into features. The classification features can, for example, be based simply on the number of matched categories for each event instance.

F. Level 3: Human Comments

Even though Level 2 allows for more flexibility, the annotators are still constrained by existing contexts within the documents. This problem is more concerning in case of sparse databases where the coverage of the explicit contexts is poor. Often, annotators make decisions using global knowledge acquired by aggregating evidence from various resources. This implicit knowledge inferred by annotators cannot be easily represented by highlighted words or categories and thus is captured neither by Level 1 nor Level 2. To address these issues, in Level 3 we allow human annotators to provide verbose comments that represent knowledge about generic situations. In our case study, the expert annotators provided the following comments:

- **Comment 1:** "*If the event is expressed by an entity (person, country, organization, etc.) in a subjective way (e.g., based on assumption, intention, consideration, plans), it's likely to have 'Other' modality. Therefore some contextual libraries including these subjective expressions should be constructed and utilized.*"
- **Comment 2:** "*If the event is likely to occur only under some certain condition, it's likely to have 'Other' modality. Therefore some contextual libraries including these condition expressions should be constructed and utilized.*"

Note that these comments refer to generic guidelines and provide richer knowledge. Some of the comments can be utilized to generate and improve the annotation guideline or train the annotators. However, a barrier in this setting is the translation of these comments into features. In our experiments, system developers manually reviewed and encoded these comments into richer and more generic features. To address these two specific comments, we created *contextual libraries* to cover these broad situations in Table X.

Table X
CONTEXTUAL CATEGORIES FOR LEVEL 3

Category Name	Size	Example Words/Phrases
Expression	116	"expressed", "debated", and "in talks about"
Consideration	77	"like", "consider", and "estimate"
Subjective	77	"assumed", "supposed", and "worried"
Intention	18	"in order to" and "for the purpose of"
Condition	18	"under" and "if"

Consequently, features were generated by checking whether the observed contexts (within the data) include any words/phrases in the above categories.

Table VIII
HIGHLIGHTED CONTEXTS FOR EVENTS WITH OTHER MODALITIES

Event Type	Trigger	Context Sentence with Highlighted Context (in bold/italic)	Expanded Highlighted Context (underlined)
Movement_ Transport	leave	Bush and Putin were ***scheduled to*** leave straight after their talks for the Group of Eight summit of the largest industrialised nations in Evian, France	***scheduled to*** : {plan to, plan for, …}
Conflict_ Attack	attacks	"We are ***warning Israel not to exploit this war*** against Iraq to carry out more attacks against the Palestinian people in the Gaza Strip…	
Justice_ Execute	execute	"***If*** we execute them now we can't bring them to life again should their appeals for a review be granted", said Antasari Azhar…	***If*** : {in case that, with the condition that, whenever, wherever, …}
Life_ Die	death	Indonesia ***will delay*** the execution of six convicts including an Indian on death row after five of them appealed to the Supreme Court for a second review	***will*** : {may, shall, could, would, … } ***delay***
Personnel_ End-Position	leave	Powell, the most moderate member of the Bush cabinet, said he fully agreed with the president's policy on Iraq and ***had no plans to*** leave	
Transaction_ Transfer-Money	payments	***It would be funded*** in two payments of 10 million dollars each upon preliminary and final court approval	***It would*** : {may, shall, could , …} ***be funded***
Transaction_ Transfer-Ownership	sell	The Stalinist state had developed nuclear weapons and ***hinted it may*** sell or use them, depending on US actions.	***hinted it may*** : {shall, could, would, …}
	acquire	Chief executive Andrew Harris said the company ***was likely to abandon plans to*** acquire a hotel in Sydney's Kings Cross red light district…	***was likely*** : {possibly, perhaps, probably, …} ***to abandon plans to***

G. Data and Scoring Metric

We randomly selected a data set from ACE 2005 newswire training set, which consists of 305 "Asserted" events and 305 "Other" events. Because of the data scarcity, ten-fold cross-validation was used to train and test the system.

H. Overall Performance

Table XI shows the accuracy scores when applying features derived from annotations at each level, along with the extra annotation costs compared to basic annotations, which were carefully recorded by the human annotators. The annotation costs do not include the time needed to design annotation scheme or train annotators. We also measured the performance of two human annotators who prepared the ACE 2005 training data on 28 newswire texts (the only subset that includes two-way annotations). As we can see, the system that utilized rich annotations achieved 6.4% absolute improvement over the baseline system using basic annotations, at a 99.9% confidence level according to the Wilcoxon Matched-Pairs Signed- Ranks Significance Test on each folder. We can also see that Level 2 annotation provides more significant gains with small extra cost compared to Level 1, therefore Level 2 can serve as a good trade-off between Level 1 and Level 3.

Figure 5 shows the results when using various size of training data. We can see that rich annotations consistently outperformed basic annotations. It's worth noting that rich annotations using only 25% training data can provide the same accuracy (69.08%) as basic annotations using the full training set. Overall the annotation cost can be reduced from 10 hours to only 3.5 hours.

I. Error Analysis

Analysis on remaining errors show that further advances would require: (1) certain degree of reasoning. For example, although there are many negation context words, the "*Personnel_Elect*" event in the following sentence should still be labeled as "*Asserted*" because the event did happen: "*Of course you will have input into the government but, since you are not directly* ***elected****, it would be a nonsense for you to have direct executive power*". (2) world knowledge such as authority detection of the news source. For example, the following "*Transaction_Transfer-Money*" event should be labeled as "*Other*" because the claims were made by

Table XI
OVERALL PERFORMANCE

Annotation Type		Accuracy	Extra Cost	Extra Effort
Basic Annotation	Level 0	69.02%	0%	––
Rich Annotation	Level 1	71.15%	25%	Highlight contexts
	Level 2	72.95%	5%	Provide category names based on highlighted contexts
	Level 3	70.82%	10%	Provide comments
	Level 1+2+3	75.41%	40%	All of the above

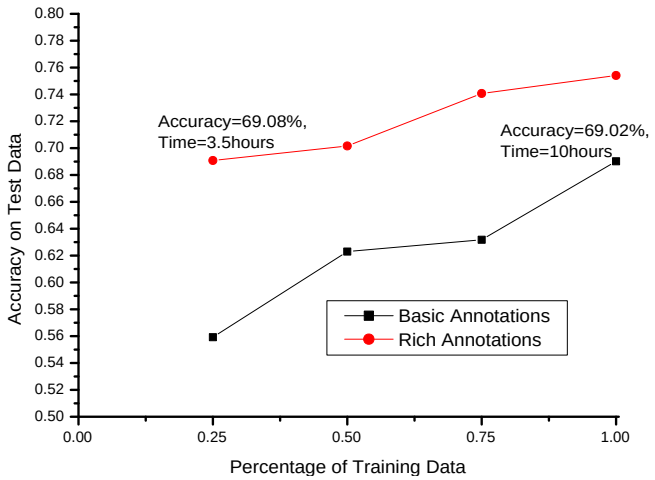

Figure 5. Impact of Training Data Size

an unauthorized source "*the suit*": "*The suit claims Iraqi officials* ***provided*** *money and training to convicted bomber Timothy McVeigh and conspirator Terry Nichols*". These are challenging cases even for human annotators.

VII. CONCLUSION AND FUTURE WORK

In a traditional supervised learning framework, a human annotator and a system are treated as isolated black-boxes to each other. We propose to better utilize the valuable knowledge from human annotators in the system development loop, by asking annotators to provide "rich annotations" for feature encoding. We investigated the trade-off between system performance and annotation cost, when adding rich annotations from various levels. We demonstrated the power and generality of this new framework on four very different case studies. The proposed framework is scalable since we measured the annotation cost on different domains. Experiments showed that the system trained from rich annotations can significantly save annotation cost in order to obtain the same performance as using basic annotations. It also outperformed some traditional validation methods, which, unlike ours, involved a great deal of feature engineering effort. The novelty of our approach lies in its declarative use of the privilege knowledge that human annotators utilize during annotation, which may address some typical errors that a system tends to make. Some of such feedback will be otherwise difficult to acquire for feature encoding (e.g., Comment 3 in name translation and Comment 4 in slot filling). On the other hand, the simplicity of our approach lies in its low cost because it incorporates the bi-product of human annotation, namely their evidence, comments and explanations, instead of tedious instance-based human correction into the learning process. In this way the human annotator's knowledge is naturally transferred to the automatic system. Hence, rich-annotation based learning is amenable to implement but pertinent to a series of common errors identified, and thus fill in the knowledge gap between human annotators and feature engineers.

Remaining error analysis suggested that our future work should focus on mining deeper world knowledge and global reasoning from annotators. Moreover, we will investigate the effects of different rich annotations provided by multiple annotators and also apply on other problem settings. In the future, we are interested in extending this idea to improve other NLP applications and integrating it with human reasoning. Ultimately we intend to investigate automatic ways to prioritize comments and convert comments to features so that we can better simulate the role of teacher in human learning.

VIII. ACKNOWLEDGEMENTS

This work was supported by the U.S. Army Research Laboratory under Cooperative Agreement No. W911NF- 09-2-0053 (NS-CTA), the U.S. NSF CAREER Award under Grant IIS-0953149, the U.S. NSF EAGER Award under Grant No. IIS- 1144111, the U.S. DARPA Broad Operational Language Translations program and PSC-CUNY Research Program. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

REFERENCES

[1] X. Li, W.-P. Lin, and H. Ji, "Comment-guided learning: Bridging the knowledge gap between expert assessor and feature engineer," in *Proc. International Conference on Advances in Information Mining and Management (IMMM2011)*, 2011.

[2] Y. Lv, L. Sun, J. Zhang, J.-Y. Nie, W. Chen, and W. Zhang, "An iterative implicit feedback approach to personalized search," in *Proc. Proc. ACL-COLING2006*, 2006.

[3] S. K. Tyler and J. Teevan, "Large scale query log analysis of re-finding," in *Proc. WSDM2010*, 2010.

[4] V. Vapnik, "Learning with teacher: Learning using hidden information," in *Proc. International Joint Conference on Neural Networks 2009*, 2009.

[5] O. F. Zaidan, J. Eisner, and C. D. Piatko, "Using annotator rationales to improve machine learning for text categorization," in *Proc. NAACL-HLT2007*, 2007.

[6] O. F. Zaidan and J. Eisner, "Modeling annotators: A generative approach to learning from annotator rationales," in *Proc. EMNLP2008*, 2008.

[7] S. Yu, F. Farooq, B. Krishnapuram, and B. Rao, "Leveraging rich annotations to improve learning of medical concepts from clinical free text," in *Proc. ICML 2011 Workshop on Learning from Unstructured Clinical Text*, 2011.

[8] H. Raghavan, O. Madani, and R. Jones, "Active learning with feedback on both features and instances," in *Journal of Machine Learning Research*, 2006.

[9] A. Haghighi and D. Klein, "Prototype-driven learning for sequence models," in *Proc. NAACL-HLT2006*, 2006.

[10] G. Druck, G. Mann, and A. McCallum, "Learning from labeled features using generalized expectation criteria," in *Proc. ACM SIGIR2008*, 2008.

[11] R. Castro, C. Kalish, R. Nowak, R. Qian, T. Rogers, and X. Zhu, "Human active learning," in *Proc. NIPS2008*, 2008.

[12] E. Bril, "Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging," in *Computational Linguistics (Volume 21, Number 1, March 1995)*, 1995.

[13] R. L. Milidiu, C. N. dos Santos, and J. C. Duarte, "Phrase chunking using entropy guided transformation learning," in *Proc. ACL-HLT2008*, 2008.

[14] L. Dini, V. D. Tornaso, and F. Segond, "Error driven word sense disambiguation," in *Proc. COLING1998*, 1998.

[15] K. Williams, C. Dozier, and A. McCulloh, "Learning transformation rules for semantic role labeling," in *Proc. CoNLL-2004*, 2004.

[16] D. B. Aronow and K. L. Coltin, "Information technology applications in quality assurance and quality improvement, part ii," *Joint Commission Journal on Quality Improvement*, vol. 10, pp. 465–478, 1993.

[17] Y. Satomura and M. B. do Amaral, "Automated diagnostic indexing by natural language processing," *Medical Informatics*, vol. 3, pp. 149–163, 1992.

[18] D. Johnson, R. Taira, W. Zhou, J. Goldin, and D. Aberle, "Hyperad, augmenting and visualizing free text radiology reports," *RadioGraphics*, vol. 18, pp. 507–515, 1998.

[19] R. Taira, S. Soderland, and R. Jakobovits, "Automatic structuring of radiology free text reports," *RadioGraphics*, vol. 21, pp. 237–245, 2001.

[20] N. Sager, M. Lyman, N. T. Nhan, and L. J. Tick, "Automatic encoding into snomed iii: A preliminary investigation," *Journal of the American Medical Informatics Association*, pp. 230–234, 1994.

[21] S. Soderland, D. Aronow, D. Fisher, J. Aseltine, and W. Lehnert, "Machine learning of text analysis rules for clinical records," *CIIR Technical Report, University of Massachusetts Amherst*, 1995.

[22] X. Zhou, H. Han, I. Chankai, A. Prestrud, and A. Brooks, "Approaches to text mining for clinical medical records," *Proceedings of the 2006 ACM Symposium on Applied Computing (Dijon, France, April 23 - 27, 2006). SAC '06. ACM, New York, NY*, pp. 235–239, 2006.

[23] A. Roberts, R. Gaizauskas, and M. Hepple, "Extracting clinical relationships from patient narratives," *BioNLP 2008: Current Trends in Biomedical Natural Language Processing. Columbus, Ohio, USA.*, pp. 10–18, 2008.

[24] C. Friedman and G. Hripcsak, "Natural language processing and its future in medicine: Can computers make sense out of natural language text," *Academic Medicine*, vol. 74, no. 8, pp. 890–895, 1999.

[25] L. Hirschman, J. C. Park, J. Tsujii, L. Wong, and C. H. Wu, "Accomplishments and challenges in literature data mining for biology," *Bioinformatics*, vol. 18, no. 12, pp. 1553–1561, 2002.

[26] W. Chapman, W. Bridewell, P. Hanbury, G. Cooper, and B. Buchanan, "Evaluation of negation phrases in narrative clinical reports," *Proceedings of the American Medical Informatics Association (AMIA) Symposium*, pp. 105–109, 2001.

[27] A. Coden, G. Savova, I. Sominsky, M. Tanenblatt, J. Masanz, K. Schuler, J. Cooper, W. Guan, and P. C., "Automatically extracting cancer disease characteristics from pathology reports into a disease knowledge representation model," *Biomedical Informatics*, 2009.

[28] G. Savova, K. Kipper-Schuler, J. Buntrock, and C. Chute, "Uima-based clinical information extraction system," *Language Resources and Evaluation: LREC: Towards enhanced interoperability for large HLT systems: UIMA for NLP*, 2008.

[29] Y. Al-Onaizan and K. Knight, "Translating named entities using monolingual and bilingual resources," in *Proc. ACL2002*, 2002.

[30] F. Huang, S. Vogel, and A. Waibel, "Improving named entity translation combining phonetic and semantic similarities," in *Proc. HLT/NAACL2004*, 2004.

[31] R. Sproat, T. Tao, and C. Zhai, "Named entity transliteration with comparable corpora," in *ACL*, 2006.

[32] A. Klementiev and D. Roth, "Named entity transliteration and discovery from multilingual comparable corpora," in *HLT-NAACL*, 2006.

[33] D. Feng, Y. Lv, and M. Zhou, "A new approach for english-chinese named entity alignment," in *Proc. PACLIC*, 2004.

[34] T. Kutsumi, T. Yoshimi, K. Kotani, and I. Sata, "Integrated use of internal and external evidence in the alignment of multi-word named entities," in *Proc. PACLIC*, 2004.

[35] R. Udupa, K. Saravanan, A. Kumaran, and J. Jagarlamudi, "Mint: A method for effective and scalable mining of named entity transliterations from large comparable corpora," in *EACL*, 2009.

[36] H. Ji, "Mining name translations from comparable corpora by creating bilingual information networks," in *ACL-IJCNLP 2009 workshop on Building and Using Comparable Corpora*.

[37] P. Fung and L. Y. Yee, "An ir approach for translating new words from nonparallel and comparable texts," in *COLING-ACL*, 1998.

[38] R. Rapp, "Automatic identification of word translations from unrelated english and german corpora," in *ACL*, 1999.

[39] L. Shao and H. T. Ng, "Mining new word translations from comparable corpora," in *COLING2004*, 2004.

[40] M. Lu and J. Zhao, "Multi-feature Based Chinese-English Named Entity Extraction from Comparable Corpora," in *Proc. PACLIC*, 2006.

[41] A. Hassan, H. Fahmy, and H. Hassan, "Improving Named Entity Translation by Exploiting Comparable and Parallel Corpora," in *Proc. RANLP2007*, 2007.

[42] A. E. Richman and P. Schone, "Mining Wiki Resources for Multilingual Named Entity Recognition," in *Proc. ACL*, 2008.

[43] E. Adar, M. Skinner, and D. S. Weld, "Information arbitrage across multi-lingual wikipedia," in *Proc. WSDM2009*.

[44] G. Bouma, S. Duarte, and Z. Islam, "Cross-lingual alignment and completion of wikipedia templates," in *The Third International Workshop on Cross Lingual Information Access: Addressing the Information Need of Multilingual Societies*, 2009.

[45] G. de Melo and G. Weikum, "Untangling the cross-lingual link structure of wikipedia," in *48th Annual Meeting of the Association for Computational Linguistics (ACL 2010), Uppsala, Sweden*, 2010.

[46] R. Navigli and S. P. Ponzetto, "Babelnet: Building a very large multilingual semantic network."

[47] W.-P. Lin, M. Snover, and H. Ji, "Unsupervised Language-Independent Name Translation Mining from Wikipedia Infoboxes," in *Proc. EMNLP2011 Workshop on Unsupervised Learning for NLP*, 2011.

[48] D. Lin, S. Zhao, B. V. Durme, and M. Pasca, "Mining parenthetical translations from the web by word alignment," in *Proc. ACL2008*, 2008.

[49] G. You, S. Hwang, Y. Song, L. Jiang, and Z. Nie, "Mining name translations from entity graph mapping," in *Proc. EMNLP2010*, 2010.

[50] L. Bentivogli, P. Clark, I. Dagan, H. Dang, and D. Giampiccolo, "The sixth pascal recognizing textual entailment challenge," in *Proc. TAC 2010 Workshop*, 2010.

[51] Z. Chen, S. Tamang, A. Lee, X. Li, W.-P. Lin, M. Snover, J. Artiles, M. Passantino, and H. Ji, "Cuny-blender tac-kbp2010 entity linking and slot filling system description," in *Proc. TAC 2010 Workshop*, 2010.

[52] V. Castelli, R. Florian, and D. jung Han, "Slot filling through statistical processing and inference rules," in *Proc. TAC 2010 Workshop*, 2010.

[53] Z. Chen and H. Ji, "A pairwise coreference model, feature impact and evaluation for event coreference resolution." in *Proc. RANLP 2009 workshop on Events in Emerging Text Types*, 2009.

[54] V. Prabhakaran, M. Bloodgood, M. Diab, B. J. Dorr, L. Levin, C. Piatko, O. Rambow, and B. V. Durme, "Statistical modality tagging from rule-based annotations and crowdsourcing," in *Proc. ACL Workshop on Extra-propositional aspects of meaning in computational linguistics (ExProM)*, 2012.

[55] R. Sauri and J. Pustejovsky, "Factbank: A corpus annotated with event factuality," in *Language Resources and Evaluation*, 2009.

[56] R. Rosales, F. Farooq, B. Krishnapuram, S. Yu, and G. Fung, "Automated identification of medical concepts and assertions in medical text," in *Proceedings of AMIA*, 2010.

[57] F. Damerau, "A technique for computer detection and correction of spelling errors," in *Communications of the ACM*, 1964.

[58] V. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," in *Soviet Physics Doklady*, 1966.

[59] H. Ji, D. Westbrook, and R. Grishman, "Using semantic relations to refine coreference decisions," in *Proc. HLT/EMNLP 05*, 2005.

[60] H. Ji, R. Grishman, H. T. Dang, K. Griffitt, and J. Ellis, "Overview of the tac 2010 knowledge base population track," in *Proc. TAC 2010 Workshop*, 2010.

[61] H. Ji, R. Grishman, and H. T. Dang, "An Overview of the TAC2011 Knowledge Base Population Track," in *Proc. Text Analytics Conference (TAC2011)*, 2011.

[62] M. Richardson and P. Domingos, "Markov logic networks," in *Machine Learning*, 2006.

[63] K. Bollacker, R. Cook, and P. Tufts, "Freebase: A shared database of structured general human knowledge," in *Proc. National Conference on Artificial Intelligence (Volume 2)*, 2007.

[64] R. Grishman, D. Westbrook, and A. Meyers, "Nyu's english ace 2005 system description," in *Proc. ACE2005*, 2005.

[65] L. Bentivogli, P. Clark, I. Dagan, H. Dang, and D. Giampiccolo, "The seventh pascal recognizing textual entailment challenge," in *Proc. TAC 2011 Workshop*, 2011.

[66] A. Penas, A. Rodrigo, V. Sama, and F. Verdejo, “Testing the reasoning for question answering validation,” in *Journal of Logic and Computation*, 2007.