Corpus Creation for New Genres:
A Crowdsourced Approach to PP Attachment

Mukund Jha, Jacob Andreas, Kapil Thadani, Sara Rosenthal and **Kathleen McKeown**
Department of Computer Science
Columbia University
New York, NY 10027, USA
{mj2472,jda2129}@columbia.edu, {kapil,sara,kathy}@cs.columbia.edu

Abstract

This paper explores the task of building an accurate prepositional phrase attachment corpus for new genres while avoiding a large investment in terms of time and money by crowdsourcing judgments. We develop and present a system to extract prepositional phrases and their potential attachments from ungrammatical and informal sentences and pose the subsequent disambiguation tasks as multiple choice questions to workers from Amazon's Mechanical Turk service. Our analysis shows that this two-step approach is capable of producing reliable annotations on informal and potentially noisy blog text, and this semi-automated strategy holds promise for similar annotation projects in new genres.

1 Introduction

Recent decades have seen rapid development in natural language processing tools for parsing, semantic role-labeling, machine translation, etc., and much of this success can be attributed to the study of statistical techniques and the availability of large annotated corpora for training. However, the performance of these systems is heavily dependent on the domain and genre of their training data, i.e. systems trained on data from a particular domain tend to perform poorly when applied to other domains and adaptation techniques are not always able to compensate (Dredze et al., 2007). For this reason, achieving high performance on new domains and genres frequently necessitates the collection of annotated training data from those domains and genres, a time-consuming and frequently expensive process.

This paper examines the problem of collecting high-quality annotations for new genres with a focus on time and cost efficiency. We explore the well-studied but non-trivial task of prepositional phrase (PP) attachment and describe a semi-automated system for identifying accurate attachments in blog data, which is frequently noisy and difficult to parse. PP attachment disambiguation involves finding a correct attachment for a prepositional phrase in a sentence. For example, in the sentence "*We went to John's house on Saturday*", the phrase "*on Saturday*" attaches to the verb "*went*". In another example, "*We went to John's house on 12th Street*", the PP "*on 12th street*" attaches to the noun "*John's house*". This sort of disambiguation requires semantic knowledge about sentences that is difficult to glean from their surface form, a problem which is compounded by the informal nature and irregular vocabulary of blog text.

In this work, we investigate whether crowdsourced human judgments are capable of distinguishing appropriate attachments. We present a system that simplifies the attachment problem and represents it in a format that can be intuitively tackled by humans.

Our approach to this task makes use of a heuristic-based system built on a shallow parser that identifies the likely words or phrases to which a PP can attach. To subsequently select the correct attachment, we leverage human judgments from multiple untrained annotators (referred to here as *workers*) through Amazon's Mechanical Turk [1], an online marketplace for work. This two-step approach of-

[1] http://www.mturk.amazon.com

Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, pages 13–20,
Los Angeles, California, June 2010.

fers distinct advantages: the automated system cuts down the space of potential attachments effectively with little error, and the disambiguation task can be reduced to small multiple choice questions which can be tackled quickly and aggregated reliably.

The remainder of this paper focuses on the PP attachment task over blog text and our analysis of the resulting aggregate annotations. We note, however, that this type of semi-automated approach is potentially applicable to any task which can be reliably decomposed into independent judgments that untrained annotators can tackle (e.g., quantifier scoping, conjunction scope). This work is intended as an initial step towards the development of efficient hybrid annotation tools that seamlessly incorporate aggregate human wisdom alongside effective algorithms.

2 Related Work

Identifying PP attachments is an essential task for building syntactic parse trees. While this task has been studied using fully-automated systems, many of them rely on parse tree output for predicting potential attachments (Ratnaparkhi et al., 1994; Yeh and Vilain, 1998; Stetina and Nagao, 1997; Zavrel et al., 1997). However, systems that rely on good parses are unlikely to perform well on new genres such as blogs and machine translated texts for which parse tree training data is not readily available.

Furthermore, the predominant dataset for evaluating PP attachment is the RRR dataset (Ratnaparkhi et al., 1994) which consists of PP attachment cases from the Wall Street Journal portion of the Penn Treebank. Instead of complete sentences, this dataset consists of sets of the form $\{V,N_1,P,N_2\}$ where $\{P,N_2\}$ is the PP and $\{V,N_1\}$ are the potential attachments. This simplification of the PP attachment task to a choice between two alternatives is unrealistic when considering the potential long-distance attachments encountered in real-world text.

While blogs and other web text, such as discussion forums and emails, have been studied for a variety of tasks such as information extraction (Hong and Davison, 2009), social networking (Gruhl et al., 2004), and sentiment analysis (Leshed and Kaye, 2006), we are not aware of any previous efforts to gather syntactic data (such as PP attachments) in the genre. Syntactic methods such as POS tagging, parsing and structural disambiguation are commonly used when analyzing well-structured text. Including the use of syntactic information has yielded improvements in accuracy in speech recognition (Chelba and Jelenik, 1998; Collins et al., 2005) and machine translation (DeNeefe and Knight, 2009; Carreras and Collins, 2009). We anticipate that datasets such as ours could be useful for such tasks as well.

Amazon's Mechanical Turk (MTurk) has become very popular for manual annotation tasks and has been shown to perform equally well over labeling tasks such as affect recognition, word similarity, recognizing textual entailment, event temporal ordering and word sense disambiguation, when compared to annotations from experts (Snow et al., 2008). While these tasks were small in scale and intended to demonstrate the viability of annotation via MTurk, it has also proved effective in large-scale tasks including the collection of accurate speech transcriptions (Gruenstein et al., 2009). In this paper we explore a method for corpus building on a large scale in order to extend annotation into new domains and genres.

We previously evaluated crowdsourced PP attachment annotation by using MTurk workers to reproduce PP attachments from the Wall Street Journal corpus (Rosenthal et al., 2010). The results demonstrated that MTurk workers are capable of identifying PP attachments in newswire text, but the approach used to generate attachment options is dependent on the existing gold-standard parse trees and cannot be used on corpora where parse trees are not available. In this paper, we build on the semi-automated annotation principle while avoiding the dependency on parsers, allowing us to apply this technique to the noisy and informal text found in blogs.

3 System Description

Our system must both identify PPs and generate a list of potential attachments for each PP in this section. Figure 1 illustrates the structure of the system.

First, the system extracts sentences from scraped blog data. Text is preprocessed by stripping HTML tags, advertisements, non-Latin and non-printable

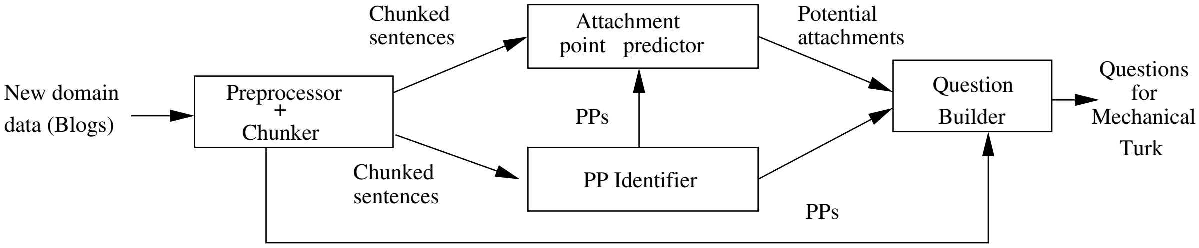


Figure 1: Overview of question generation system

characters. Emoticon symbols are removed using a standard list. [2]

The cleaned data is then partitioned into sentences using the NLTK sentence splitter. [3] In order to compensate for the common occurrence of informal punctuation and web-specific symbols in blog text, we replace all punctuation symbols between quotation marks and parentheses with placeholder tags (e.g. ⟨QuestionMark⟩) during the sentence splitting process and do the same for website names, time markers and referring phrases (e.g. *@John*). Additionally, we attempt to re-split sentences at ellipsis boundaries if they are longer than 80 words and discard them if this fails.

As parsers trained on news corpora tend to perform poorly on unstructured texts like blogs, we rely on a chunker to partition sentences into phrases. Choosing a good chunker is essential to this approach: around 35% of the cases in which the correct attachment is not predicted by the system are due to chunker error. We experimented with different chunkers over a random sample of 50 sentences before selecting a CRF-based chunker (Phan, 2006) for its robust performance.

The chunker output is initially processed by fusing together chunks in order to ensure that a single chunk represents a complete attachment point. Two consecutive NP chunks are fused if the first contains an element with a possessive part of speech tag (e.g. *John's book*), while particle chunks (PRT) are fused with the VP chunks that precede them (e.g. *pack up*). These chunked sentences are then processed to identify PPs and potential attachment points for them, which can then be used to generate questions for MTurk workers.

3.1 PP Extraction

PPs can be classified into two broad categories based on the number of chunks they contain. A *simple PP* consists of only two chunks: a preposition and one noun phrase, while a *compound PP* has multiple simple PPs attached to its primary noun phrase. For example, in the sentence "*I just made some last-minute changes to the latest issue of our newsletter*", the PP with preposition "*to*" can be considered to be either the simple PP "*to the latest issue*" or the compound PP "*to the latest issue of our newsletter*".

We handle compound PPs by breaking them down into multiple simple PPs; compound PPs can be recovered by identifying the attachments of their constituent simple PPs. Our simple PP extraction algorithm identifies PPs as a sequence of chunks that consist of one or more prepositions terminating in a noun phrase or gerund.

3.2 Attachment Point Prediction

A PP usually attaches to the noun or verb phrase preceding it or, in some cases, can modify a following clause by attaching to the head verb. We build a set of rules based on this intuition to pick out the potential attachments in the sentence; these rules are described in Table 1. The rules are applied separately for each PP in a sentence and in the same sequence as mentioned in the table (except for rule 4, which is applied while choosing a chunk using any of the other rules).

[2] http://www.astro.umd.edu/~marshall/smileys.html

[3] http://www.nltk.org

	Rule	**Example**
1	Choose closest NP and VP preceding the PP.	I made modifications **to our newsletter**.
2	Choose next closest VP preceding the PP if the VP selected in (1) contains a VBG.	He snatched the disk flying away **with one hand**.
3	Choose first VP following the PP.	**On his desk** he has a photograph.
4	All chunks inside parentheses are skipped, unless the PP falls within parentheses.	Please refer to the new book (second edition) **for more notes**.
5	Choose anything immediately preceding the PP that is not out of chunk and has not already been picked.	She is full **of excitement**.
6	If a selected NP contains the word *and*, expand it into two options, one with the full expression and one with only the terms following *and*.	He is president and chairman **of the board**.
7	For PPs in chains of the form P-NP-P-NP (PP-PP), choose all the NPs in the chain preceding the PP and apply all the above rules considering the whole chain as a single PP.	They found my pictures of them **from the concert**.
8	If there are fewer than four options after applying the above rules, also select the VP preceding the last VP selected, the NP preceding the last NP selected, and the VP following the last VP picked.	

Table 1: List of rules for attachment point predictor. In the examples, PPs are denoted by boldfaced text and potential attachment options are underlined.

4 Experiments

An experimental study was undertaken to test our hypothesis that we could obtain reliable annotations on informal genres using MTurk workers. Here we describe the dataset and our methods.

4.1 Dataset and Interface

We used a corpus of blog posts made on LiveJournal [4] for system development and evaluation. Only posts from English-speaking countries (i.e. USA, Canada, UK, Australia and New Zealand) were considered for this study.

The interface provided to MTurk workers showed the sentence on a plain background with the PP highlighted and a statement prompting them to pick the phrase in the sentence that the given PP modified. The question was followed by a list of options. In addition, we provided MTurk workers the option to indicate problems with the given PP or the listed options. Workers could write in the correct attachment if they determined that it wasn't present in the list of options, or the correct PP if the one they were presented with was malformed. This allowed them to correct errors made by the chunker and automated attachment point predictor. In all cases, workers were forced to pick the best answer among the options regardless of errors. We also supplied a number of examples covering both well-formed and erroneous cases to aid them in identifying appropriate attachments.

4.2 Experimental Setup

For our experiment, we randomly selected 1000 questions from the output produced by the system and provided each question to five different MTurk workers, thereby obtaining five different judgments for each PP attachment case. Workers were paid four cents per question and the average completion time per task was 48 seconds. In total $225 was spent on the full study with $200 spent on the workers and $25 on MTurk fees.The total time taken for the study was approximately 16 hours.

A pilot study was carried out with 50 sentences before the full study to test the annotation interface and experiment with different ways of presenting the PP and attachment options to workers. During this study, we observed that while workers were willing to suggest correct answers or PPs when faced with erroneous questions, they often opted to not pick any of the options provided unless the question was well-formed. This was problematic because, in many cases, expert annotators *were* able to identify the most appropriate attachment option. Therefore, in the final study we forced them to pick the most suitable option from the given choices before indicating errors and writing in alternatives.

[4] http://www.livejournal.com

Workers in agreement	Number of questions	Accuracy	Coverage
5 (unanimity)	389	97.43%	41.33%
≥ 4 (majority)	689	94.63%	73.22%
≥ 3 (majority)	887	88.61%	94.26%
≥ 2 (plurality)	906	87.75%	96.28%
Total	**941**	**84.48%**	**100%**

Table 2: Accuracy and coverage over agreement thresholds

5 Evaluation corpus

In order to determine if the MTurk results were reliable, worker responses had to be validated by having expert annotators perform the same task. For this purpose, two of the authors annotated the 1000 questions used for the experiment independently and compared their judgments. Disagreements were observed in 127 cases; these were then resolved by a pool of non-author annotators. If all three annotators on a case disagreed with each other the question was discarded; this situation occured 43 times. An additional 16 questions were discarded because they did not have a valid PP. For example, "*I am painting with my blanket on today*". Here "*on today*" is incorrectly extracted as a PP because the particle "on" is tagged as a preposition. The rest of the analysis presented in this section was performed on the remaining 941 sentences.

The annotators' judgments were compared to the answers provided by the MTurk workers and, in the case of disagreement between the experts and the majority of workers, the sentences were manually inspected to determine the reason. In five cases, more than one valid attachment was possible; for example, in the sentence "*The video below is of my favourite song on the album - A Real Woman*", the PP "*of my favourite song*" could attach to either the noun phrase "*the video*" or the verb "*is*" and conveys the same meaning. In such cases, both the experts and the workers were considered to have chosen the correct answer.

In 149 cases, the workers also augmented their choices by providing corrections to incomplete answers and badly constructed PPs. For example, the PP "*of the Rings and Mikey*" in the sentence "*Samwise from Lord of the Rings and Mikey from The Goonies are the same actor ?*" was corrected to "*of the Rings*". In 34/39 of the cases where the correct answer was not present in the options provided, at least one worker indicated correct attachment for the PP.

5.1 Attachment Prediction Evaluation

We measure the recall for our attachment point predictor as the number of questions for which the correct attachment appeared among the generated options divided by the total number of questions. The system achieves a recall of 95.85% (902/941 questions). We observed that in many cases where the correct attachment point was not predicted, it was due to a chunker error. For example, in the following sentence, "*Stop all the clocks , cut off the telephone , Prevent the dog from barking with a juicy bone...*", the PP "*from barking*" attaches to the verb "*Prevent*"; however, due to an error in chunking "*Prevent*" is tagged as a noun phrase and hence is not picked by our system. The correct attachment was also occasionally missed when the attachment point was too far from the PP. For example, in the sentence "*Fitting as many people as possible on one sofa and under many many covers and getting intimate*", the correct attachment for the PP "*under many many covers*" is the verb "*Fitting*" but it is not picked by our system.

Even though the correct attachment was not always given, the workers could still provide their own correct answer. In the first example above, 3/5 workers indicated that the correct attachment was not in the list of options and wrote it in.

6 Results

Table 2 summarizes the results of the experiment. We assess both the coverage and reliability of worker predictions at various levels of worker agreement. This serves as an indicator of the effectiveness of the MTurk results: the accuracy can be taken

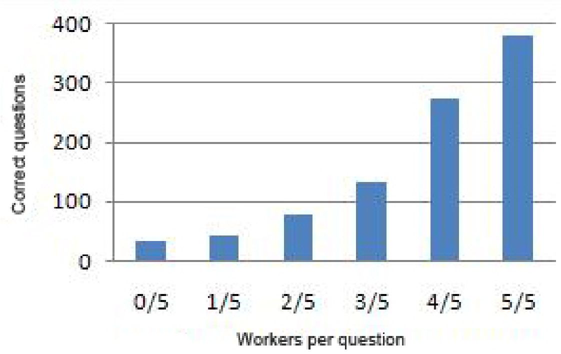


Figure 2: The number of questions in which exactly x workers provided the correct answer

as a general confidence measure for worker predictions; when five workers agree we can be 97.43% confident in the correctness of their prediction, when at least four workers agree we can be 94.63% confident, etc. *Unanimity* indicates that all workers agreed on an answer, *majority* indicates that more than half of workers agreed on an answer, and *plurality* indicates that two workers agreed on a single answer, while the remaining three workers each selected different answers. We observe that at high levels of worker agreement, we get extremely high accuracy but limited coverage of the data set; as we decrease our standard for agreement, coverage increases rapidly while accuracy remains relatively high.

Figure 2 shows the number of workers providing the correct answer on a per-question basis. This illustrates the distribution of worker agreements across questions. Note that in the majority of cases (69.2%), at least four workers provided the correct answer; in only 3.6% of cases were no workers able to select the correct attachment.

Figure 3 shows the distribution of worker agreements. Unlike Table 2, these figures are not cumulative and include non-plurality two-worker agreements. Note that the number of agreements discussed in this figure is greater than the 941 evaluated because in some cases there were multiple agreements on a single question. As an example, three workers may choose one answer while the remaining two workers choose another; this question then produces both a three-worker agreement as well as a two-worker agreement.

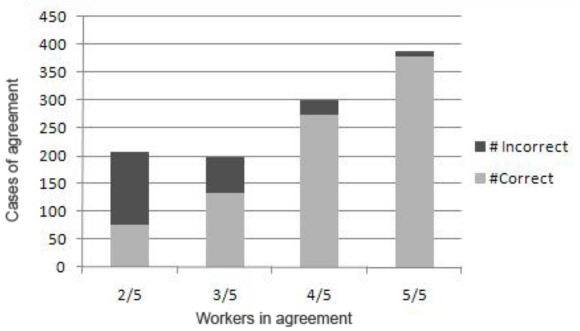


Figure 3: The number of cases in which exactly x workers agreed on an answer

No. of options	No. of cases	Accuracy
< 4	179	86.59%
4	718	84.26%
> 4	44	79.55%

Table 3: Variation in worker performance with the number of attachment options presented

All questions on which there is agreement also produce a majority vote, with one exception: the 2/2/1 agreement. Although the correct answer was selected by one set of two workers in every case of 2/2/1 agreement, this is not particularly useful for corpus-building as we have no way to identify *a priori* which set is correct. Fortunately, 2/2/1 agreements were also quite rare and occurred in only 3% of cases.

Figure 3 appears to indicate that instances of agreement between two workers are unlikely to produce good attachments; they have a an average accuracy of 37.2%. However, this is due in large part to cases of 3/2 agreement, in which the two workers in the minority are usually wrong, as well as cases of 2/2/1 agreement which contain at least one incorrect instance of two-worker agreement. However, if we only consider cases in which the two-worker agreement forms a plurality (i.e. all other workers disagree amongst themselves), we observe an average accuracy of 64.3% which is similar to that of cases of three-worker agreement (67.7%).

We also attempted to study the variation in worker performance based on the complexity of the task; specifically looking at how response accuracy varied depending on the number of options that workers were presented with. Although our system aimed to

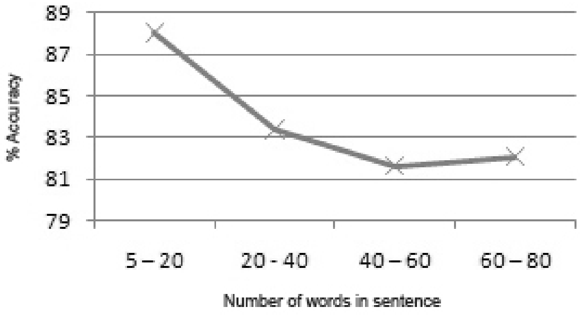


Figure 4: Variation in accuracy with sentence length.

generate four attachment options per case, fewer options were produced for small sentences and opening PPs while additional options were generated in sentences containing PP-NP chains (see Table 1 for the complete list of rules). Table 3 shows the variation in accuracy with the number of options provided to the workers. We might expect that an increased number of options may be correlated with decreased accuracy and the data does indeed seem to suggest this trend; however, we do not have enough datapoints for the cases with fewer or more than four options to verify whether this effect is significant.

We also analyzed the relationship between the length of the sentence (in terms of number of words) and the accuracy. Figure 4 indicates that as the length of the sentence increases, the average accuracy decreases. This is not entirely unexpected as lengthy sentences tend to be more complicated and therefore harder for human readers to parse.

7 Conclusions and Future Work

We have shown that by working in conjunction with automated attachment point prediction systems, MTurk workers are capable of annotating PP attachment problems with high accuracy, even when working with unstructured and informal blog text. This work provides an immediate framework for the building of PP attachment corpora for new genres without a dependency on full parsing.

More broadly, the semi-automated framework outlined in this paper is not limited to the task of annotating PP attachments; indeed, it is suitable for almost any syntactic or semantic annotation task where untrained human workers can be presented with a limited number of options for selection. By dividing the desired annotation task into smaller sub-tasks that can be tackled independently or in a pipelined manner, we anticipate that more syntactic information can be extracted from unstructured text in new domains and genres without the sizable investment of time and money normally associated with hiring trained linguists to build new corpora. To this end, we intend to further leverage the advent of crowdsourcing resources in order to tackle more sophisticated annotation tasks.

Acknowledgements

The authors would like to thank Kevin Lerman for his help in formulating the original ideas for this work. This material is based on research supported in part by the U.S. National Science Foundation (NSF) under IIS-05-34871. 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 NSF.

References

Xavier Carreras and Michael Collins. 2009. Non-projective parsing for statistical machine translation. In *Proceedings of EMNLP*, pages 200–209.

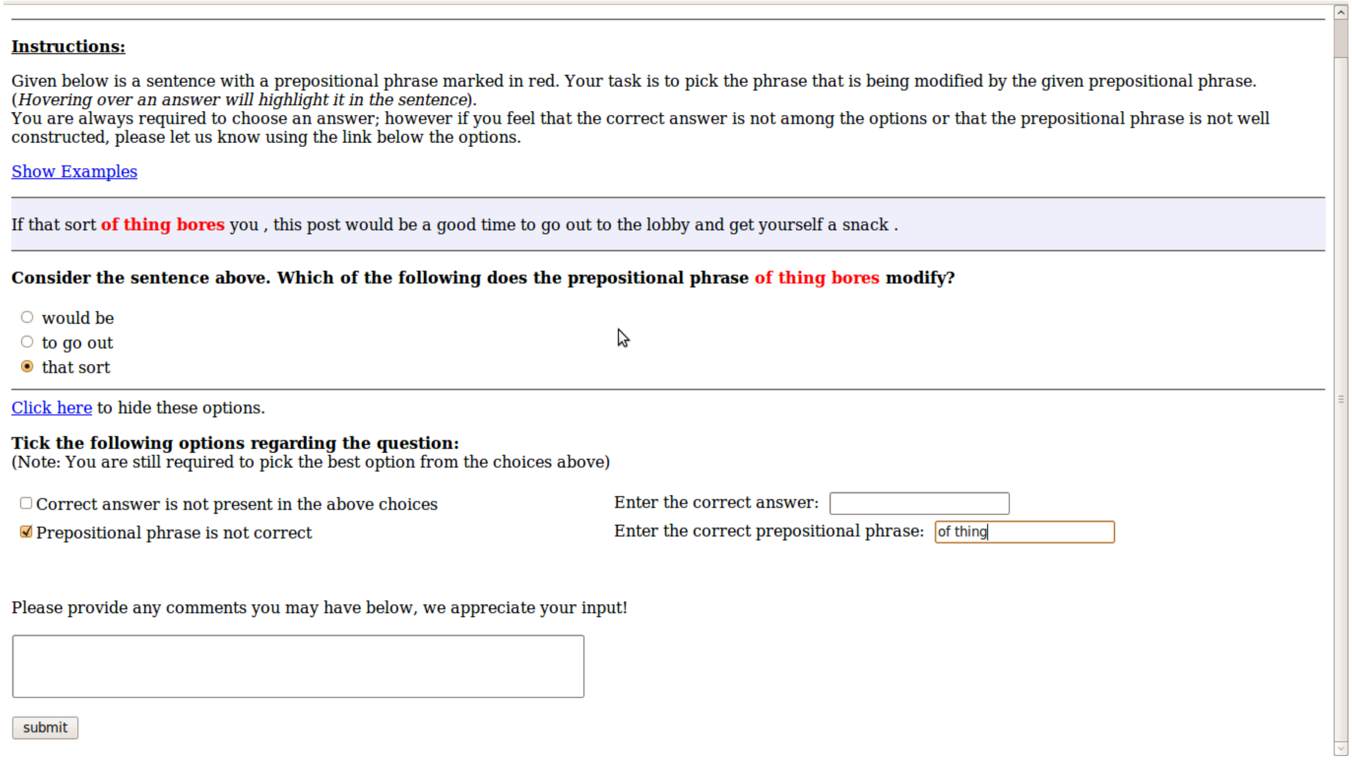
Ciprian Chelba and Frederick Jelenik. 1998. Structured language modeling for speech recognition. In *Proceedings of NLDB*.

Michael Collins, Brian Roark, and Murat Saraclar. 2005. Discriminative syntactic language modeling for speech recognition. In *Proceedings of ACL*, pages 507–514.

Steve DeNeefe and Kevin Knight. 2009. Synchronous tree adjoining machine translation. In *Proceedings of EMNLP*, pages 727–736.

Mark Dredze, John Blitzer, Partha Pratim Talukdar, Kuzman Ganchev, João Graca, and Fernando Pereira. 2007. Frustratingly hard domain adaptation for dependency parsing. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL*, pages 1051–1055, Prague, Czech Republic, June. Association for Computational Linguistics.

Alex Gruenstein, Ian McGraw, and Andrew Sutherland. 2009. A self-transcribing speech corpus: collecting continuous speech with an online educational game. In *Proceedings of the Speech and Language Technology in Education (SLaTE) Workshop*.



Instructions:

Given below is a sentence with a prepositional phrase marked in red. Your task is to pick the phrase that is being modified by the given prepositional phrase. (*Hovering over an answer will highlight it in the sentence*).
You are always required to choose an answer; however if you feel that the correct answer is not among the options or that the prepositional phrase is not well constructed, please let us know using the link below the options.

Show Examples

If that sort **of thing bores** you , this post would be a good time to go out to the lobby and get yourself a snack .

Consider the sentence above. Which of the following does the prepositional phrase of thing bores modify?

○ would be
○ to go out
● that sort

Click here to hide these options.

Tick the following options regarding the question:
(Note: You are still required to pick the best option from the choices above)

□ Correct answer is not present in the above choices	Enter the correct answer:
☑ Prepositional phrase is not correct	Enter the correct prepositional phrase: of thing

Please provide any comments you may have below, we appreciate your input!

Figure 5: HIT Interface for PP attachment task

Daniel Gruhl, R. Guha, David Liben-Nowell, and Andrew Tomkins. 2004. Information diffusion through blogspace. In *Proceedings of WWW*, pages 491–501.

Liangjie Hong and Brian D. Davison. 2009. A classification-based approach to question answering in discussion boards. In *Proceedings of SIGIR*, pages 171–178.

Gilly Leshed and Joseph 'Jofish' Kaye. 2006. Understanding how bloggers feel: recognizing affect in blog posts. In *CHI '06 extended abstracts on Human factors in computing systems*, pages 1019–1024.

Xuan-Hieu Phan. 2006. CRFChunker: CRF English phrase chunker. http://crfchunker.sourceforge.net.

Adwait Ratnaparkhi, Jeff Reynar, and Salim Roukos. 1994. A maximum entropy model for prepositional phrase attachment. In *Proceedings of HLT*, pages 250–255.

Sara Rosenthal, William J. Lipovsky, Kathleen McKeown, Kapil Thadani, and Jacob Andreas. 2010. Semi-automated annotation for prepositional phrase attachment. In *Proceedings of LREC*, Valletta, Malta.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. 2008. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of EMNLP*, pages 254–263.

Jiri Stetina and Makoto Nagao. 1997. Corpus based PP attachment ambiguity resolution with a semantic dictionary. In *Proceedings of the Workshop on Very Large Corpora*, pages 66–80.

Alexander S. Yeh and Marc B. Vilain. 1998. Some properties of preposition and subordinate conjunction attachments. In *Proceedings of COLING*, pages 1436–1442.

Jakub Zavrel, Walter Daelemans, and Jorn Veenstra. 1997. Resolving PP attachment ambiguities with memory-based learning. In *Proceedings of the Workshop on Computational Language Learning (CoNLL)*, pages 136–144.

Appendix A: Mechanical Turk Interface

Figure 5 shows a screenshot of the interface provided to the Mechanical Turk workers for the PP attachment task. By default, examples and additional options are hidden but can be viewed using the links provided. The screenshot illustrates a case in which a worker is confronted with an incorrect PP and uses the additional options to correct it.