# An Example-Based Approach to Difficult Pronoun Resolution

**Canasai Kruengkrai** **Naoya Inoue** **Jun Sugiura** **Kentaro Inui**
Graduate School of Information Sciences, Tohoku University
6-6 Aramaki Aza Aoba, Aobaku, Sendai, Miyagi 980-8579, Japan
{canasai,naoya-i,jun-s,inui}@ecei.tohoku.ac.jp

## Abstract

A Winograd schema is a pair of twin sentences containing a referential ambiguity that is easy for a human to resolve but difficult for a computer. This paper explores the characteristics of world knowledge necessary for resolving such a schema. We observe that people tend to avoid ambiguous antecedents when using pronouns in writing. We present a method for automatically acquiring examples that are similar to Winograd schemas but have less ambiguity. We generate a concise search query that captures the essential parts of a given source sentence and then find the alignments of the source sentence and its retrieved examples. Our experimental results show that the existing sentences on the Web indeed contain instances of world knowledge useful for difficult pronoun resolution.

## 1 Introduction

Consider the following pair of sentences:[1]

(1) a. The outlaw shot the sheriff, but *he* did not shoot the deputy.
b. The outlaw shot the sheriff, but *he* shot back.

Suppose that the target pronoun is *he*, and its two candidate antecedents are *the outlaw* and *the sheriff*. The question is which of the two candidates is the correct antecedent for the target pronoun in each sentence? Most people resolve *he* to *the outlaw* in (1a) but to *the sheriff* in (1b) without noticing any ambiguity. However, for a computer program, this pronoun resolution becomes extremely difficult, requiring the use of world knowledge and the ability to reason. We refer to the pair of sentences like (1) as a *Winograd schema* (Levesque, 2011; Levesque et al., 2012). Note that the two sentences differ only in a few words and have a referential ambiguity that is resolved in opposite ways.

A previous work by Rahman and Ng (2012) showed that two sources of world knowledge, including narrative chains (Chambers and Jurafsky, 2008) and page counts returned by a search engine, are useful for resolving Winograd schemas. However, these two knowledge sources have their own weaknesses and need some heuristics to bridge the gap. Narrative chains suffer from the lack of discourse relations. For example, both sentences in (1) have a contrast relation indicated by *but*. However, narrative chains rely only on temporal relations between two events (e.g., *before* and *after*). Page counts used for estimating $n$-gram statistics are unstable and vary considerably over time (Lapata and Keller, 2005; Levesque et al., 2012). Therefore the answer to the question "what kind of world knowledge does a computer program need to have to resolve Winograd schemas?" (Levesque, 2013) is still unclear.

Rather than looking for new knowledge bases, we first examine whether existing sentences on the Web have sufficient evidence that could be applied to resolve Winograd schemas. If such evidence is available, we may be able to later generalize a collection of those sentences into a more abstract level of representation.

[1]The sentences are taken from the dataset created by Rahman and Ng (2012).


28th Pacific Asia Conference on Language, Information and Computation pages 358–367

This paper explores the characteristics of world knowledge necessary for resolving Winograd schemas. We observe that people tend to avoid ambiguous antecedents when using pronouns in writing. Consider the following sentences derived from Web snippets:

(2) a. I shot Sherry, but I did not shoot Debbie.
b. Deputy Daniel Russ was working security outside the busy courthouse and was shot in the leg, but he shot back.

Both sentences in (2) have less ambiguity and are easier to be resolved. A vanilla coreference resolver can predict the coreference chains denoted by the underlined words in each sentence. Note that *he* in (2b) who shot back is the subject, while *Deputy Daniel Russ* who was shot is the object. Based on the structural similarity between (1b) and (2b), we infer that *he* in (1b) should be resolved to *the sheriff*, which is also the object. Likewise, *he* in (1a) should be resolved to *the outlaw* using the clue from (2a).

We present a method for automatically acquiring examples that are similar to Winograd schemas but have less ambiguity. First, we generate a concise search query that captures the essential parts of a given source sentence. Then, we find the alignments of the source sentence and its retrieved examples. Finally, we rank the most likely antecedent for the target pronoun using our score function.

In the following section, we discuss related work. Section 3 presents our approach. Section 4 shows our experimental results and error analysis. Section 5 concludes the paper with some directions of future research.

## 2 Related work

We classify the problem of pronoun resolution into two main categories: traditional anaphora and Winograd schemas.

Anaphora (or coreference) resolution has a long history in NLP. Ng (2010) and Poesio et al. (2011) provided excellent surveys of approaches to anaphora resolution. A variety of corpora and evaluation metrics also made it difficult for researchers to compare the performance of their systems. To establish benchmarking data and evaluation metrics, the CoNLL-2011 and CoNLL-2012 shared tasks mainly focused on coreference resolution (Pradhan et al., 2011; Pradhan et al., 2012).

The term "Winograd schema" was coined by Hector Levesque (2011), named after Terry Winograd who first used a pair of twin sentences to show the difficulty in natural language understanding (Winograd, 1972). Levesque proposed the Winograd Schema (WS) Challenge as an alternative to the Turing Test, which aims to test artificially intelligent systems. Unlike the Turing Test, the WS Challenge just requires systems to answer a collection of binary questions. These questions called Winograd schemas are pairs of sentences containing referential ambiguities that are easy for people to resolve but difficult for systems. A Winograd schema is designed to satisfy the following constraints (Levesque et al., 2012):

- Easily disambiguated by people;
- Not solvable by simple linguistic techniques;
- No obvious statistics over text corpora.

Levesque (2011) first provided an initial set of 19 Winograd schemas.[2] Rahman and Ng (2012) later released a relaxed version of Winograd schemas, consisting of 941 examples constructed by undergraduate students. In general, a WS sentence has main and subordinate clauses. The main clause has two candidate antecedents, and the subordinate clause has a target pronoun. The task is to resolve the target pronoun to one of the two candidate antecedents.

Shallow semantic attributes (e.g., gender and number) and grammatical relations would be useful for the traditional anaphora resolution. However, these linguistic features are not sufficient to solve the WS Challenge. Rahman and Ng (2012) proposed a ranking-based model that combines sophisticated linguistic features derived from different sources of world knowledge, such as narrative chains (Chambers and Jurafsky, 2008) and page counts returned by Google. Narrative chains are built by considering temporal relations between two events. However, the WS Challenge contains various discourse

[2] A collection of Winograd schemas has been updated and is available at: http://www.cs.nyu.edu/davise/papers/WS.html.

relations, such as explanation and contrast. Balasubramanian et al. (2013) found another issue of narrative chains in which unrelated actors are often mixed into the same chains. Lapata and Keller (2005) and Levesque et al. (2012) examined the use of page counts and found the stability issue.

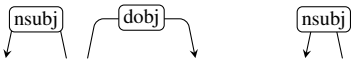
The contribution of our work is the exploration of the necessary background knowledge for resolving the WS Challenge. To better understand the nature of the WS sentences, we propose to examine similar sentences having less ambiguity and develop a method for automatically acquiring those similar sentences from the Web.

## 3 Approach

Our goal is to acquire useful examples that are similar to the WS sentences. We try to retain lexical, syntactic, semantic, and discourse properties of the WS sentences. We represent a source sentence using the Stanford dependency (Section 3.1) and generate a search query to acquire examples from the Web (Section 3.2). We then align pairs of the source sentence and its retrieved examples (Section 3.3) and rank the most likely antecedent for the target pronoun using our score function (Section 3.4).

### 3.1 Dependency representation

We need to transform a sentence to a more generalized structure. The Stanford dependency (SD) representation is a practical scheme (de Marneffe et al., 2006). A dependency captures a grammatical relation holding between a head and a dependent. All dependencies for the sentence then map onto a directed graph, where words in the sentence are nodes and grammatical relations are edge labels. For example, focusing on dependencies for the candidate antecedents and the target pronoun, the sentence (1b) has the dependency structure as follows:




(3) The outlaw shot the sheriff, but he shot back.

In the main clause, the subject and direct object of $shot_3$ are $outlaw_2$ and $sheriff_5$, respectively. In the subordinate clause, the subject of $shot_9$ is $he_8$. The subscript indicates the word position in the sentence, including punctuation. Note that we only use headwords of candidate antecedents determined by using the Collins head rules (Collins, 1999). For example, the headword of the noun phrase "the sheriff" is "sheriff".

| | | | |
|---|---|---|---|
| because (310) | that (16) | however (2) | until (2) |
| but (82) | even though (4) | as (2) | after (2) |
| since (69) | if (3) | then (2) | hence (1) |
| so (46) | although (2) | what (2) | |
| and (15) | when (2) | out of (2) | |

Table 1: Statistics of conjunctions in Rahman's test set.

### 3.2 Example acquisition

We use the Google Web Search API to acquire examples from the Web. We consider Google's snippets as sentences and try to extract examples from these snippets. The question is what kind of examples would be useful for resolving difficult pronouns? Here we expect that a good example should have linguistic properties similar to a given source sentence but has less ambiguity. For example, the examples (2a) and (2b) have the similar grammatical, semantic, and discourse relations to the source sentences (1a) and (1b), but their pronouns are easier to be resolved. To retrieve such examples, our search queries should capture the essential parts of the source sentences while still being concise. In what follows, we describe our criteria on how to retain words in the source sentence when generating a search query.

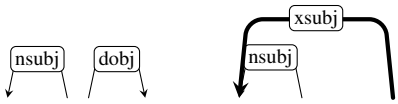
**Conjunction** A WS sentence contains two clauses connected with a conjunction. The conjunction reflects a discourse relation between the two clauses. A line of work in cognitive science and linguistics shows that the discourse relation has a strong influence on pronoun interpretation (Hobbs, 1979; Kehler et al., 2008; Rohde and Kehler, 2013). Therefore a useful example should have the same discourse relation as the source sentence. Table 1 shows the statistics of conjunctions in Rahman's test set. The majority of discourse relations are explanation (e.g., *because* and *since*), followed by contrast (e.g., *but*).[3]

**Heads of actors** The two candidate antecedents and the target pronoun act certain roles in the source sentence. We capture their roles through the SD

[3]In our experiments, we used *because* as the representative word for *since* when generating the search query.

representation. For example, in (3), *outlaw*$_2$ and *sheriff*$_5$ serve as the subject and direct object of their head *shot*$_3$, while *he*$_8$ functions as the subject of its head *shot*$_9$. We then keep these two heads, *shot*$_3$ and *shot*$_9$, as well as the conjunction *but*$_7$.

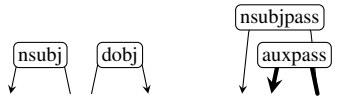
In the SD representation, a word can have multiple heads. For example, consider the following sentence:




(4) Paper beats rock, but it is able to beat scissors.

The heads of *it*$_6$ are *able*$_8$ and *beat*$_{10}$, where *nsubj* and *xsubj* denote the nominal subject and the controlling subject, respectively. In the case of multiple heads, we only keep the rightmost head, *beat*$_{10}$.

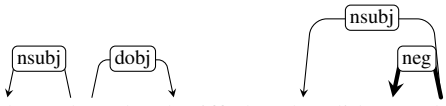
**Verb to be** In the SD representation, a copula verb like *be* is treated as an auxiliary modifier (de Marneffe and Manning, 2008). For example, consider the following sentence, which is a twin of (4):




(5) Paper beats rock, but it is beaten by scissors.

Focusing on the subordinate clause, we first keep *beaten*$_8$ which is the head of *it*$_6$. The auxiliary *is*$_7$ is also important since it helps to indicate the passive form of *beaten*$_8$. Therefore we also keep the verb to be if it is the auxiliary modifier of the head.

**Negation** Negation is an important grammatical operation since it can invert the meaning of the clause or sentence. For example, omitting negation in (6) could make the whole sentence difficult to understand. Therefore we also keep the negation modifier of the head:




(6) The outlaw shot the sheriff, but he did not shoot the deputy.

**Dependent of a light head** A head of an actor could be a light verb, which is a verb that has little meaning on its own. For example, consider the following sentence:


(7) The lion bit the gazelle, because it had sharp teeth.

Based on our criteria, we first keep the heads of the actors and the conjunction, including *bit*$_3$, *because*$_7$, and *had*$_9$. However, the lemma form of *had*$_9$ is a light verb, which does not adequately explain the reason for *bit*$_3$. Therefore we also keep the dependent of the light head, *teeth*$_{11}$, to make explanation more clear. In our experiments, we defined {*be*, *do*, *have*, *make*} as a set of the light verbs. In the case of multiple dependents, we only select the rightmost one.

**Phrasal verb particle** A particle after a verb often provides a specific meaning to that verb. For example, "shot back" in (1b) indicates a reaction against the action of the main clause. Therefore we also keep the particle following the head.

In summary, given a source sentence, we keep the conjunction and the heads of the two candidate antecedents and the target pronoun. We then check the dependents of the heads, keeping only those that meet our criteria. We replace other words with asterisks. Multiple consecutive asterisks are combined into one. For example, we generate the search queries for (1a) and (1b) as follows:

(8) a. *"* shot * but * not shoot *"*
b. *"* shot * but * shot back"*

and for (4) and (5) as:

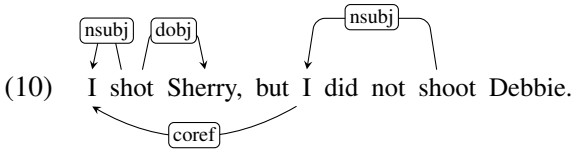
(9) a. *"* beats * but * is beaten by *"*
b. *"* beats * but * beat *"*

### 3.3 Alignment

After retrieving snippets, we analyze them using the Stanford CoreNLP (Manning et al., 2014). We use the standard pipeline, ranging from tokenization to dependency parsing. A snippet may contain several fragments or sentences, so we consider it as a short document. We then use the Berkeley coreference resolver (Durrett and Klein, 2013) for predicting coreference chains within each snippet. We consider the processed snippets as candidate examples. For example, (2a) has the following coreference chain:

| Relation | Description |
|---|---|
| *subject* | |
| *nsubj* | nominal subject |
| *xsubj* | controlling subject |
| *csubj* | clausal subject |
| *agent* | agent |
| *object* | |
| *dobj* | direct object |
| *iobj* | indirect object |
| *pobj* | object of preposition |
| *nsubjpass* | passive nominal subject |

Table 2: Generalized grammatical relations.




We also experimented with the Stanford coreference resolver but found that the Berkeley resolver is more robust to noisy text. We discuss the characteristics of these two resolvers in Section 4.2.

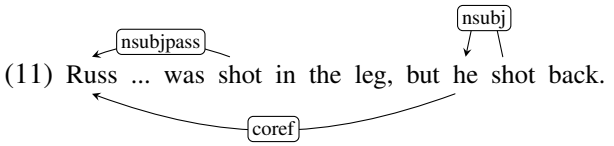
Next, we try to find alignments of a source sentence and its candidate examples. Our scheme is simple. The source sentence and the candidate example is an alignment if they satisfy the following conditions:

- The heads of the actors are synonymous.
- The grammatical roles of the heads are in the same category.

Note that the Google Web Search API expands some queries and returns results containing related words. As a result, we use the synonym instead of the exact match to increase coverage.[4] We also generalize grammatical relations to a coarser level. Here we focus on two main categories: *subject* and *object*. Table 2 shows our generalized grammatical relations.

Based on our scheme, the dependency structures (6) and (10), where their original sentences are (1a) and (2a), are a good alignment since their heads and grammatical roles match exactly. We write an analogy in the form $A{:}B{::}C{:}D$, meaning $A$ is to $B$ as $C$ is to $D$ (Turney, 2006). Therefore we derive a candidate analogy $I_6{:}I_1{::}he_8{:}outlaw_2$ from the alignment of (6) and (10).

Consider the following dependency structure, which corresponds to (2b):




Note that we omit some words due to the limited space. Although (11) has one actor, $Russ_3$, and his grammatical role, *nsubjpass*, does not match exactly with those of the actors in (3), the dependency structures (3) and (11), where their original sentences are (1b) and (2b), are still a good alignment since the grammatical roles *nsubjpass* and *dobj* are in the same *object* category. Therefore we obtain a candidate analogy $he_{19}{:}Russ_3{::}he_8{:}sheriff_5$ from the alignment of (3) and (11).

### 3.4 Ranking candidate antecedents

We use candidate analogies to rank the two candidate antecedents for the target pronoun in a given source sentence. The target pronoun is resolved to a higher scoring antecedent. A simple score function is to count the number of candidate analogies of each antecedent. Note that our alignments are based on automatic processing of snippet texts, inevitably containing an amount of noise. So we would like to distinguish between acceptable and good alignments.

Let us introduce some notation. A source sentence $i$ contains a target pronoun $p_i$ and its two candidate antecedents $a_{i,k}$, $k \in \{1, 2\}$. An example $j$ contains a pronoun $p_j$ and its predicted antecedent $a_j$. We write $p_j{:}a_j{::}p_i{:}a_{i,k}$ for an analogy of the alignment of $j$ and $i$. We define the score of a candidate antecedent $a_{i,k}$ as the sum of the scores of all candidate analogies: $\sum_j \text{score}(p_j{:}a_j{::}p_i{:}a_{i,k})$. We then apply the attributional similarity for factoring the score of each candidate analogy (Turney, 2006). Our score function becomes:

$$\text{score}(p_j{:}a_j{::}p_i{:}a_{i,k}) = \frac{1}{2}(\text{s}_a(p_j, p_i) + \text{s}_a(a_j, a_{i,k})).$$

Finally, we estimate the attributional similarity $\text{s}_a$ by augmenting the similarity of the heads $h$ of the

[4] We use WordNet in Natural Language Toolkit (Bird et al., 2009).

corresponding dependencies:

$$\mathrm{s}_a(p_j, p_i) = d(p_j, p_i) + d(h(p_j), h(p_i)) \,,$$

where $d$ is the path distance similarity of two word senses available in Natural Language Toolkit (Bird et al., 2009).[5] We estimate $\mathrm{s}_a(a_j, a_{i,k})$ using the same fashion. For example, we compute the score of the analogy $I_6$:$I_1$::$he_8$:$outlaw_2$ derived from the alignment of (6) and (10) as: $\frac{1}{2}(d(I_6, he_8) + d(shoot_9, shoot_{11}) + d(I_1, outlaw_2) + d(shot_2, shot_3)) = \frac{1}{2}(0.33 + 1.0 + 0.09 + 1.0) = 1.21$.

## 4 Experiments

### 4.1 Dataset and setting

We used the dataset created by Rahman and Ng (2012).[6] Their dataset can be viewed as a relaxed version of Winograd schemas since the target pronouns in some sentences could be resolved using selectional restrictions. For example, consider the following sentence: "*Lions eat zebras because they are predators*". The counts returned by Google for "*lions are predators*" are significantly higher than those of "*zebras are predators*". In other words, the system could resolve *they* to *lions* without considering the relationship between two clauses. Note that our approach does not use this kind of counting in resolving the difficult pronouns.

Our approach is a pure example-based strategy, which requires no training data. Therefore we only use Rahman's test set. In the following experiments, we only considered the test sentences where the grammatical roles of the actors are in the coarse-grained subject or object categories (Table 2), and the two candidate antecedents share the same head. For example, in (3), *outlaw*$_2$ and *sheriff*$_5$ share the same head *shot*$_3$. We retained 244 out of the original 564 test sentences.

Next, we generated search queries for these test sentences. Accessing the Google Web Search API is not trivial since the number of requests is limited for free use. We paused 20 seconds between each query and retrieved only top two pages (8 results per page). Therefore the maximum number of results for a given query is 16. We also tried to increase the number of retrieved pages but found that lower ranked pages tend to be irrelevant. In this stage, we obtained results for 185 (out of 244) queries and no results for 59 queries. For example, the search query "** sued * because * was embezzling*" generated from "*Bob sued Bill because he was embezzling funds*" received no results since these terms have not explicitly co-occurred in Google's database.

After extracting examples from snippets and aligning, 155 (out of 185) test sentences could be aligned with at least one example. Some examples did not contain either coreference chains or compatible dependencies. We refer to the remaining 155 test sentences as **D1**. To ensure that each test sentence has a twin, we also generated a subset of D1 containing 120 test sentences denoted by **D2**. In the case of D2, if a system uniformly resolves the target pronoun to the subject (or object), it can achieve 50% accuracy.

### 4.2 Baselines and evaluation metrics

We also conducted experiments using existing coreference resolvers to see whether they could handle the difficult pronouns. We experimented with two publicly available resolvers and our baseline system:

**STANFORD** is the winner of the CoNLL-2011 shared task (Raghunathan et al., 2010; Lee et al., 2011). STANFORD is a rule-based system that applies precision-ordered sieves (filtering rules) to decide whether two mentions should be linked. For noun-pronoun mention pairs, STANFORD first assigns semantic attributes to the mentions. The semantic attributes include number, gender, animacy, and NER labels, which are derived from existing knowledge sources (Bergsma and Lin, 2006; Ji and Lin, 2009; Finkel et al., 2005). STANFORD links two mentions if their attributes have no disagreement.

**BERKELEY** is the current state-of-the-art coreference resolution system based on the mention-ranking approach (Durrett and Klein, 2013). BERKELEY learns to link two mentions using surface features that capture linguistic properties of mentions and mention pairs. BERKELEY also inherits semantic attributes from STANFORD and uses them as shallow semantic features. In

[5] Note that a word can have many senses. So we iterate over the Cartesian product of two synsets and use the maximum similarity score.

[6] http://www.hlt.utdallas.edu/~vince/data/emnlp12

| System | D1 Correct | D1 Incorrect | D1 No Decision | D2 Correct | D2 Incorrect | D2 No Decision |
|---|---|---|---|---|---|---|
| STANFORD | 45.16% (70/155) | 45.16% (70/155) | 9.68% (15/155) | 46.67% (56/120)) | 46.67% (56/120) | 6.67% (8/120) |
| BERKELEY$_{pre}$ | 49.68% (77/155) | 49.68% (77/155) | 0.65% (1/155) | 50.00% (60/120) | 50.00% (60/120) | 0.00% (0/120) |
| BERKELEY$_{new}$ | 50.32% (78/155) | 48.39% (75/155) | 1.29% (2/155) | 50.83% (61/120) | 49.17% (59/120) | 0.00% (0/120) |
| MENTRANKER | 55.48% (86/155) | 44.52% (69/155) | 0.00% (0/155) | 54.17% (65/120) | 45.83% (55/120) | 0.00% (0/120) |
| OURS | 69.68% (108/155) | 29.68% (46/155) | 0.65% (1/155) | 72.50% (87/120) | 27.50% (33/120) | 0.00% (0/120) |

Table 3: Experimental results on the D1 and D2 test sets.

our experiments, we used the pre-trained model (BERKELEY$_{pre}$) as well as retrained a new model (BERKELEY$_{new}$) using Rahman's training set.[7]

**MENTRANKER** is our baseline mention ranker. We tried to replicate the ranking-based model described in Rahman and Ng (2012). We explored five features, including narrative chains,[8] Google, semantic compatibility, heuristic polarity, and lexical features. Note that some of our knowledge sources are different from those of Rahman and Ng (2012). For Google, we used the counts from the Google $n$-gram dataset (Brants and Franz, 2006). For semantic compatibility, instead of using BLLIP, Reuters, and English Gigaword, we extracted the features from the ClueWeb12 dataset.[9]

We provided gold mentions (the two candidate antecedents and the target pronoun) as the inputs for each baseline system in testing. Therefore the baseline systems did not need to perform mention detection. For evaluation, we followed Rahman and Ng (2012). Given a test sentence, the system could *correctly*, *incorrectly*, or *not* resolve the target pronouns.

### 4.3 Results

Table 3 shows our experimental results. The shallow semantic attributes used in STANFORD do not seem to be helpful for resolving the difficult pronouns. STANFORD also left many sentences unresolved. For example, consider the following sentences:

(12) a. Lions love gazelles because *they* eat them.
b. Lions love gazelles because *they* are delicious.

The two candidate antecedents (*lions* and *gazelles*) are animate and plural, which can be compatible with the target pronoun *they*.

The surface features used in BERKELEY$_{pre}$ are also not helpful for handling the difficult pronouns. Retraining BERKELEY$_{new}$ with Rahman's training set has almost no impact. Here we do not intend to indicate that STANFORD and BERKELEY are ineffective in general. We would rather say that the shallow semantic features used in the coreference literature are not sufficient for resolving the difficult pronouns. MENTRANKER exploits more sophisticated features extracted from different knowledge sources. However, MENTRANKER performs slightly better than STANFORD and BERKELEY.[10]

Our approach acquires examples from the Web and uses them to facilitate decision. For example, the following examples were retrieved and applied for resolving (12):

(13) a. I love Easter because I get to eat lots of chocolate.
b. I love them because they are delicious and the whole family likes them.

While (13a) supports resolving *they* to *lions* in (12a), (13b) helps resolving *they* to *gazelles* in (12b). Our approach correctly resolves 69.68% and 72.50% of the target pronouns in D1 and D2, respectively.

### 4.4 Error analysis

We manually examined errors made by our approach. We found that a common source of errors is due to automatic processing of the data, such as parsing and predicting coreference chains in snippet

[7] We parsed Rahman's training set using the Stanford CoreNLP, converted it to the CoNLL format, and retrained a new model using the 'trainOnGold' option, which yielded better results in our experiments.

[8] http://www.usna.edu/users/cs/nchamber/data/schemas/acl09

[9] http://www.lemurproject.org/clueweb12

[10] Rahman and Ng (2012) showed that narrative chains yield improved accuracy for resolving the WS Challenge. However, the improvement comes not only from narrative chains but also from other (unintentionally added) features (personal communication).

| | |
|---|---|
| (14) | Sally gave **Kelly** a doll because *she* loved dolls.<br>I gave you that power because I loved you and trusted you completely<br>He gave his life a ransom, just because he loved me so |
| (15) | Mary gave **Sandy** her book because *she* needed it.<br>I gave Mike Branch a call because I needed some help with a trailer loading problem<br>I only gave $16.00, because I needed change and needed to decide to give them how much tips |
| (16) | The cat broke **the glass** because *it* was fragile.<br>the glass broke because it was fragile<br>I broke down crying because I was so fragile<br>If the toilet broke from a light touch because it was so fragile the landlord would pay |
| (17) | **The cat** broke the glass because *it* was clumsy.<br>In this story the donkey broke the manger because he was clumsy |
| (18) | Olga kicked **Sara** because *she* woke her up.<br>I kicked Zayn because I woke up on the wrong side of the bed<br>I could have kicked myself because I woke up late |
| (19) | **Olga** kicked Sara because *she* was drunk.<br>he kicked her out of Homecoming dance because she was drunk in the parking lot<br>he got kicked off because he was drunk at rehearsals |
| (20) | The coach told **the captain** that *he* was fired.<br>Williams told Fox News that he was fired Wednesday by Ellen Weiss, NPR's vice president for news<br>When I applied for unemployment benefits, I was honest and told them that I was fired |

Table 4: Samples of errors made by our approach. In each row, the first line is the source sentence followed by its examples. In each source sentence, the correct antecedent is boldfaced and the target pronoun is italicized. In each example, the coreferent mentions are underlined.

texts. We also inspected some of errors based on the scores of incorrectly resolved antecedents. An incorrect antecedent with a large score gap means that most retrieved examples support the opposite antecedent to the answer. Examples of this kind of errors are shown in Table 4. In what follows, we discuss some interesting linguistic phenomena observed from the errors.

**Direct and indirect objects** The source sentences (14) and (15) have the same pattern. The main clause has the *subject-transfer_verb-indirect_object-direct_object* pattern, where the verb *gave* is a transfer verb. In the subordinate clause, the target pronoun interacts with *direct_object* (e.g., "she loved dolls"). In their corresponding examples, the target pronoun instead interacts with *indirect_object* (e.g., "I loved you") or has no interaction. One solution for this case is to use predefined patterns to eliminate irrelevant examples. However, the utility of such patterns is quite limited.

**Selectional restrictions** In the source sentence (16), the adjective *fragile* seems to co-occur more frequently with *glass* than *cat*. In the source sentence (17), the subject of the adjective *clumsy* is more likely to be an animate noun (e.g., *cat*) than an inanimate noun (e.g., *glass*). The use of selectional restrictions could be helpful for handling such cases in Rahman's dataset. Note that, in (17), our baseline coreference resolver, BERKELEY, incorrectly resolved *he* to *manger*, which is an inanimate noun.

**Transitive and intransitive verbs** The verb *broke* is used as a transitive verb (e.g., "the cat broke the glass") in the source sentence (16) but as an intransitive verb (e.g., "the glass broke" and "I broke down crying") in its examples. Likewise, in (18), the phrasal verb *woke up* is used in different functions. Distinguishing between the transitive and intransitive verbs could be a useful feature.

**No obvious answer** In the source sentence (19), the antecedent was chosen by using the background knowledge that someone who was drunk tends to do bad things. Since *Olga* was drunk, *she* should be the one who kicks other people. However, the opposite answer is possible. As in the corresponding examples, someone who was drunk can be punished by being kicked.

**Semantic relation between actors** The source sentence (20) was constructed by using the background knowledge that the noun *coach* has a higher status than the noun *captain* in a team environment. In other words, someone who has a higher status can fire other people. Note that the answer can be flipped if the two nouns are replaced with proper names.

## 5 Conclusion

We have only scratched the surface of the most fundamental question "what kind of world knowledge does a computer program need to have to pass the WS Challenge?" (Levesque, 2013). We explore the necessary background knowledge for resolving the WS Challenge. Our key observation is that people tend to avoid ambiguous antecedents when using pronouns in writing. We present a method for automatically acquiring examples that are similar to Winograd schemas but have less ambiguity. We generate a concise search query that captures the essential parts of a given source sentence and then find the alignments of the source sentence and its retrieved examples. Our experimental results show that the existing sentences on the Web indeed contain instances of world knowledge useful for difficult pronoun resolution.

Our current approach has several limitations. We only considered the WS sentences in which the actors have specific grammatical roles and share the same head. We plan to examine other sentence structures. For example, consider the following sentence: "*Lakshman asked Vivan to get him some ice cream because he was hot*". In this case, *asked* is the head of *Lakshman*, while *get* is the head of *Vivan*. We also plan to handle the WS sentences that have no obvious examples.

Our error analysis reveals that resolving the WS Challenge requires not only a wide range of world knowledge but also expressive representations that can handle the complexities of natural language. There is a line of research that tries to map natural language sentences to formal semantic representations (Kamp and Reyle, 1993; Steedman, 2000; Copestake et al., 2005; Liang et al., 2011; Banarescu et al., 2013). Exploring the usefulness of these semantic representations would be an important direction for future work.

## References


Niranjan Balasubramanian, Stephen Soderland, Mausam, and Oren Etzioni. 2013. Generating coherent event schemas at scale. In *Proceedings of EMNLP*, pages 1721–1731.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *Proceedings of Linguistic Annotation Workshop*.

Shane Bergsma and Dekang Lin. 2006. Bootstrapping path-based pronoun resolution. In *Proceedings of ACL*, pages 33–40.

Steven Bird, Edward Loper, and Ewan Klein. 2009. *Natural Language Processing with Python*. O'Reilly Media Inc.

Thorsten Brants and Alex Franz. 2006. Web 1t 5-gram version 1. https://catalog.ldc.upenn.edu/LDC2006T13.

Nathanael Chambers and Daniel Jurafsky. 2008. Unsupervised learning of narrative event chains. In *Proceedings of ACL*, pages 789–797.

Michael Collins. 1999. Head-driven statistical models for natural language parsing. *Ph.D. thesis*.

Ann Copestake, Dan Flickinger, Ivan Sag, and Carl Pollard. 2005. Minimal recursion semantics: An introduction. *Research on Language and Computation*, 3:281–332.

Marie-Catherine de Marneffe and Christopher D. Manning. 2008. Stanford dependencies manual. http://nlp.stanford.edu/software/stanford-dependencies.shtml.

Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC*, pages 449–454.

Greg Durrett and Dan Klein. 2013. Easy victories and uphill battles in coreference resolution. In *Proceedings of EMNLP*.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of ACL*, pages 363–370.

Jerry R. Hobbs. 1979. Coherence and coreference. *Cognitive Science*, 3:67–90.

Heng Ji and Dekang Lin. 2009. Gender and animacy knowledge discovery from web-scale n-grams for unsupervised person mention detection. In *Proceedings of PACLIC*.

Hans Kamp and Uwe Reyle. 1993. *From Discourse to Logic: An Introduction to the Model-theoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Kluwer, Dordrecht.

Andrew Kehler, Laura Kertz, Hannah Rohde, and Jeffrey L. Elman. 2008. Coherence and coreference revisited. *Journal of Semantics*, 25:1–44.

Mirella Lapata and Frank Keller. 2005. Web-based models for natural language processing. *ACM Trans. Speech Lang. Process.*, 2(1).

Heeyoung Lee, Yves Peirsman, Angel Chang, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2011. Stanford's multi-pass sieve coreference resolution system at the CoNLL-2011 shared task. In *Proceedings of CoNLL: Shared Task*, pages 28–34.

Hector Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *Proceedings of Principles of Knowledge Representation and Reasoning*.

Hector Levesque. 2011. The winograd schema challenge. In *Commonsense*.

Hector Levesque. 2013. On our best behaviour. *IJCAI Research Excellence Award Presentation*.

Percy Liang, Michael I. Jordan, and Dan Klein. 2011. Learning dependency-based compositional semantics. In *Proceedings of ACL*, pages 590–599.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of ACL: System Demonstrations*, pages 55–60.

Vincent Ng. 2010. Supervised noun phrase coreference research: The first fifteen years. In *Proceedings of ACL*, pages 1396–1411.

Massimo Poesio, Simone Paolo Ponzetto, and Yannick Versley. 2011. Computational models of anaphora resolution: A survey. *Linguistic Issues in Language Technology*.

Sameer Pradhan, Lance Ramshaw, Mitchell Marcus, Martha Palmer, Ralph Weischedel, and Nianwen Xue. 2011. CoNLL-2011 shared task: Modeling unrestricted coreference in ontonotes. In *Proceedings of CoNLL: Shared Task*, pages 1–27.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In *Proceedings of EMNLP/CoNLL: Shared Task*, pages 1–40.

Karthik Raghunathan, Heeyoung Lee, Sudarshan Rangarajan, Nathanael Chambers, Mihai Surdeanu, Dan Jurafsky, and Christopher Manning. 2010. A multi-pass sieve for coreference resolution. In *Proceedings of EMNLP*, pages 492–501.

Altaf Rahman and Vincent Ng. 2012. Resolving complex cases of definite pronouns: The winograd schema challenge. In *Proceedings of EMNLP-CoNLL*, pages 777–789.

Hannah Rohde and Andrew Kehler. 2013. A probabilistic reconciliation of coherence-driven and centering-driven theories of pronoun interpretation. *Theoretical Linguistics*, 39:1–37.

Mark Steedman. 2000. *The Syntactic Process*. MIT Press.

Peter D. Turney. 2006. Similarity of semantic relations. *Computational Linguistics*, 32(3):379–416.

Terry Winograd. 1972. *Understanding Natural Language*. Academic Press, Inc.