Resolving Shell Nouns

Varada Kolhatkar
Department of Computer Science
University of Toronto
Toronto, ON, M5S 3G4, Canada
varada@cs.toronto.edu

Graeme Hirst
Department of Computer Science
University of Toronto
Toronto, ON, M5S 3G4, Canada
gh@cs.toronto.edu

Abstract

Shell nouns, such as *fact* and *problem*, occur frequently in all kinds of texts. These nouns themselves are unspecific, and can only be interpreted together with the *shell content*. We propose a general approach to automatically identify shell content of shell nouns. Our approach exploits lexico-syntactic knowledge derived from the linguistics literature. We evaluate the approach on a variety of shell nouns with a variety of syntactic expectations, achieving accuracies in the range of 62% (baseline = 33%) to 83% (baseline = 74%) on crowd-annotated data.

1 Introduction

Shell nouns are abstract nouns, such as *fact*, *issue*, *idea*, and *problem*, which facilitate efficiency by avoiding repetition of long stretches of text. The *shell* metaphor comes from Schmid (2000), and it captures the various functions of these nouns in a discourse: containment, signalling, pointing, and encapsulating. Shell nouns themselves are unspecific, and can only be interpreted together with their *shell content*, i.e., the propositional content they encapsulate in the given context. The process of identifying this content in the given context is referred to as *shell noun resolution* or *interpretation*. Examples (1), (2), and (3) show usages of the shell nouns *fact* and *issue*. The shell noun phrases are resolved to the postnominal *that* clause, the complement *wh* clause, and the immediately preceding clause, respectively.[1,2]

(1) <u>**The fact**</u> **that a major label hadn't been at liberty to exploit and repackage the material on CD** meant that prices on the vintage LP market were soaring.

(2) <u>**The issue**</u> that this country and Congress must address is **how to provide optimal care for all without limiting access for the many**.

(3) **Living expenses are much lower in rural India than in New York**, but <u>**this fact**</u> is not fully captured if prices are converted with currency exchange rates.

Observe that the relation between shell noun phrases and their shell content is similar to the relation of abstract anaphora (or cataphora) (Asher, 1993) with backward- or forward-looking abstract-object antecedents. For anaphoric shell noun examples, the shell content precedes the shell noun phrase, and for cataphoric shell noun examples the shell content follows the shell noun phrase.[3]

Shell nouns as a group occur frequently in argumentative texts (Schmid, 2000; Flowerdew, 2003; Botley, 2006). They play an important role in organizing a discourse and maintaining its coherence (Schmid, 2000; Flowerdew, 2003), and resolving them is an important component of various computational linguistics tasks that rely on

[1] Note that the postnominal *that*-clause in (1) is not a relative clause: the fact in question is not an argument of *exploit and repackage*.

[2] All examples in this paper are from the New York Times corpus (https://catalog.ldc.upenn.edu/LDC2008T19)

[3] We use the terms cataphoric shell noun and anaphoric shell noun for lack of better alternatives.

Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 499–510,
October 25-29, 2014, Doha, Qatar.

discourse structure. Accordingly, identifying shell content can be helpful in summarization, information retrieval, and ESL learning (Flowerdew, 2003; Hinkel, 2004).

Despite their importance in discourse, understanding of shell nouns from a computational linguistics perspective is only in the preliminary stage. Recently, we proposed an approach to annotate and resolve anaphoric cases of six typical shell nouns: *fact*, *reason*, *issue*, *decision*, *question*, and *possibility* (Kolhatkar et al., 2013b). This work drew on the observation that shell nouns following cataphoric constructions are easy to resolve. We manually developed rules to identify shell content for such cases. Later, we used these cataphoric examples and their shell content as training data to resolve harder anaphoric examples.

In this paper, we propose a general algorithm to resolve cataphoric shell noun examples. Our long-term goal is to build an end-to-end shell-noun resolution system. If we want to go beyond the six shell nouns from our previous work, and generalize our approach to other shell nouns, first we need to develop an approach to resolve cataphoric shell noun examples. A number of challenges are associated with this seemingly easy task. The primary challenges is that this resolution is in many crucial respects a semantic phenomenon. To obtain the required semantic knowledge, we exploit the properties of shell nouns and their categorization described in the linguistics literature. We evaluate our method using crowdsourcing, and demonstrate how far one can get with simple, deterministic shell content extraction.

2 Related work

Shell-nounhood is a well-established concept in linguistics (Vendler, 1968; Ivanic, 1991; Asher, 1993; Francis, 1994; Schmid, 2000, inter alia). However, understanding of shell nouns from a computational linguistics perspective is only in the preliminary stage.

Shell nouns take a number of semantic arguments. In this respect, they are similar to the general class of argument-taking nominals as given in the NomBank (Meyers et al., 2004). Similarly, there is a small body of literature that addresses nominal semantic role labelling (Gerber et al., 2009) and nominal subcategorization frames (Preiss et al., 2007). That said, the distinguishing property of shell nouns is that one of their semantic arguments is the shell content, but the literature in computational linguistics does not provide any method that is able to identify the shell content. The focus of our work is to rectify this.

Shell content represents complex and abstract objects. So traditional linguistic and psycholinguistic principles used in pronominal anaphora resolution (see the survey by Poesio et al. (2011)), such as gender and number agreement, are not applicable in resolving shell nouns. That said, there is a line of literature on annotating and resolving personal and demonstrative pronouns, which typically refer to similar kinds of non-nominal abstract entities (Passonneau, 1989; Eckert and Strube, 2000; Byron, 2003; Müller, 2008; Hedberg et al., 2007; Poesio and Artstein, 2008; Navarretta, 2011, inter alia). Also, there have been attempts at annotating the shell content of anaphoric occurrences of shell nouns (e.g., Botley (2006), Kolhatkar et al. (2013a)). However, none of these approaches attempt to annotate and resolve cataphoric examples such (1) and (2).

3 Challenges

A number of challenges are associated with the task of resolving cataphoric shell noun examples, especially when it comes to developing a holistic approach for a variety of shell nouns.

First, each shell noun has idiosyncrasies. Different shell nouns have different semantic and syntactic expectations, and hence they take different types of one or more semantic arguments: one introducing the shell content, and others expressing circumstantial information about the shell noun. For instance, *fact* typically takes a single factual clause as an argument, which is its shell content, as we saw in example (1), whereas *reason* expects two arguments: the cause and the effect, with the content introduced in the cause, as shown in example (4).[4] Similarly, *decision* takes an agent making the decision and the shell content is represented as an action or a proposition, as shown in (5).[5]

(4) <u>**One reason**</u> [that 60 percent of New York City public-school children read below grade level]$^{\text{effect}}$ is [**that many elementary schools don't have libraries**]$^{\text{cause}}$.

[4]Observe that the postnominal *that* clause in (4) is not a relative clause, and still it is not the shell content because it is not the cause argument of the shell noun *reason*.

[5]Observe that this aspect of shell nouns of taking different numbers and kinds of complement clauses is similar to verbs having different subcategorization frames.

(5) I applaud loudly the decision of [Greenburgh]^{agent} **to ban animal performances**.

Second, the relation between a shell noun and its content is in many crucial respects a semantic phenomenon. For instance, resolving the shell noun *reason* to its shell content involves identifying a) that *reason* generally expects two semantic arguments: cause and effect, b) that the cause argument (and not the effect argument) represents the shell content, and c) that a particular constituent in the given context represents the cause argument.

Third, at the conceptual level, once we know which semantic argument represents shell content, resolving examples such as (4) seems straightforward using syntactic structure, i.e., by extracting the complement clause. But at the implementation level, this is a non-trivial problem for two reasons. The first reason is that examples containing shell nouns often follow syntactically complex constructions, including embedded clauses, coordination, and sentential complements. An automatic parser is not always accurate for such examples. So the challenge is whether the available tools in computational linguistics such as syntactic parsers and discourse parsers are able to provide us with the information that is necessary to resolve these difficult cases. The second reason is that the shell content can occur in many different constructions, such as apposition (e.g., ***parental ownership of children**, **a concept** that allows* …), postnominal and complement clause constructions, as we saw in examples (1) and (2), and modifier constructions (e.g., ***the liberal trade policy** that* …). Moreover, in some constructions, the content is indefinite (e.g., ***A bad idea** does not harm until someone acts upon it.*) or *None* because the example is a non-shell noun usage (e.g., ***this week's issue** of Sports Illustrated*), and the challenge is to identify such cases.

Finally, whether the postnominal clause introduces the shell content or not is dependent on the context of the shell noun phrase. The resolution can be complicated by complex syntactic constructions. For instance, when the shell noun follows verbs such as *expect*, it becomes difficult for an automatic system to identify whether the postnominal or the complement clause is of the verb or of the shell noun (e.g., *they did not expect **the decision** to reignite tension in Crown Heights* vs. *no one expected **the decision to call an election***). Similarly, shell noun phrases can be objects of prepositions, and whether the postnominal clause introduces the shell content or not is dependent on this preposition. For instance, for the pattern *reason that*, the postnominal *that* clause does not generally introduce the shell content, as we saw in (4); however, this does not hold when the shell noun phrase containing *reason* follows the preposition *for*, as shown in (6).

(6) Low tax rates give people an incentive to work, for **the simple reason that they get to keep more of what they earn**.

4 Linguistic framework

Linguists have studied a variety of shell nouns, their classification, different patterns they follow, and their semantic and syntactic properties in detail (Vendler, 1968; Ivanic, 1991; Asher, 1993; Francis, 1994; Schmid, 2000, inter alia). Schmid points out that being a shell noun is a property of a specific usage of the noun rather than an inherent property of the word. He provides a list of 670 English nouns that tend to occur as shell nouns. A few frequently occurring ones are: *problem*, *notion*, *concept*, *issue*, *fact*, *belief*, *decision*, *point*, *idea*, *event*, *possibility*, *reason*, *trouble*, *question*, *plan*, *theory*, *aim*, and *principle*.

4.1 Lexico-syntactic patterns

Precisely defining the notion of shell-nounhood is tricky. A necessary property of shell nouns is that they are capable of taking clausal arguments, primarily with two lexico-syntactic constructions: *Noun + postnominal clause* and *Noun + be + complement clause* (Vendler, 1968; Biber et al., 1999; Schmid, 2000; Huddleston and Pullum, 2002). Schmid exploits these lexico-syntactic constructions to identify shell noun usages. In particular, he provides a number of typical lexico-syntactic patterns that are indicative of either anaphoric or cataphoric shell noun occurrences. Table 1 shows these patterns with examples.

Cataphoric These patterns primarily follow two constructions.

N-be-clause In this construction, the shell noun phrase occurs as the subject in a subject-verb-clause construction, with the linking verb *be*, and the shell content embedded as a *wh* clause, *that* clause, or *to*-infinitive clause. The linking

	Cataphoric	
1	*N-be-to*	**Our plan is** ***to*** *hire and retain the best managers we can.*
2	*N-be-that*	**The major reason is** ***that*** *doctors are uncomfortable with uncertainty.*
3	*N-be-wh*	Of course, **the central, and probably insoluble, issue is** ***whether*** *animal testing is cruel.*
4	*N-to*	**The decision** ***to*** *disconnect the ventilator* came after doctors found no brain activity.
5	*N-that*	Mr. Shoval left open **the possibility** ***that*** *Israel would move into other West Bank cities.*
6	*N-wh*	If there ever is **any doubt** ***whether*** *a plant is a poppy or not*, break off a stem and squeeze it.
7	*N-of*	**The concept of** *having an outsider as Prime Minister* is outdated.
	Anaphoric	
8	*th-N*	*Living expenses are much lower in rural India than in New York*, but **this fact** is not fully captured if prices are converted with currency exchange rates.
9	*th-be-N*	*People change*. **This is a fact**.
10	*Sub-be-N*	If the money is available, however, ***cutting the sales tax*** **is a good idea**.

Table 1: Lexico-grammatical patterns of shell nouns (Schmid, 2000). Shell noun phrases are underlined, the pattern is marked in boldface, and the shell content is marked in italics.

	Proportion							
Noun	***N-be-to***	***N-be-that***	***N-be-wh***	***N-to***	***N-that***	***N-wh***	***N-of***	**total**
idea	7	2	-	5	23	10	53	91,277
issue	-	1	5	7	14	2	71	55,088
concept	1	-	-	6	12	-	79	14,301
decision	-	-	-	80	12	1	5	55,088
plan	5	-	-	72	17	-	4	67,344
policy	4	1	-	16	25	2	51	24,025

Table 2: Distribution of cataphoric patterns for six shell nouns in the New York Times corpus. Each column shows the percentage of instances following that pattern. The last column shows the total number of cataphoric instances of each noun in the corpus.

verb *be* indicates the semantic identity between the shell noun and its content in the given context. The construction follows the patterns in rows 1, 2, and 3 of Table 1.

N-clause This construction includes the cataphoric patterns 4–7 in Table 1. For these patterns the link between the shell noun and the content is much less straightforward: whether the postnominal clause expresses the shell content or not is dependent on the shell noun and the syntactic structure under consideration. For instance, for the shell noun *fact*, the shell content is embedded in the postnominal *that* clause, as shown in (1), but this does not hold for the shell noun *reason* in example (4). The *N-of* pattern is different from other patterns: it follows the construction *N-prepositional phrase* rather than *N-clause*, and since a prepositional phrase can take different kinds of embedded constituents such as a noun phrase, a sentential complement, and a verb phrase, the pattern offers flexibility in the syntactic type of the shell content.

Anaphoric For these patterns, the link between the shell noun and the content is created using linguistic elements such as *the*, *this*, *that*, *other*, *same*, and *such*. For the patterns 8 and 9 the shell content does not typically occur in the sentence containing the shell noun phrase. For the pattern 10, the shell content is the subject in a subject-verb-N construction.

Pattern preferences Different shell nouns have different pattern preferences. Table 2 shows the distribution of cataphoric patterns for six shell nouns in the New York Times corpus. The shell nouns *idea*, *issue*, and *concept* prefer *N-of* pattern, whereas *plan* and *decision* prefer the pattern *N-to*. Among all instances of the shell noun *decision* fol-

Idea family	*Plan* family
Semantic features: [mental], [conceptual]	**Semantic features**: [mental], [volitional], [manner]
Frame: mental; focus on propositional content of IDEA	**Frame**: mental; focus on IDEA
Nouns: ***idea***, ***issue***, ***concept***, *point*, *notion*, *theory*, . . .	**Nouns**: ***decision***, ***plan***, ***policy***, *idea*, . . .
Patterns: *N-be-that/of*, *N-that/of*	**Patterns**: *N-be-to/that*, *N-to/that*
Trouble **family**	***Problem*** **family**
Semantic features: [eventive], [attitudinal], [manner], [deontic]	**Semantic features**: [factual], [attitudinal], [impeding]
Frame: general eventive	**Frame**: general factual
Nouns: ***problem***, ***trouble***, ***difficulty***, *dilemma*, *snag*	**Nouns**: ***problem***, ***trouble***, ***difficulty***, *point*, *thing*, *snag*, *dilemma* , . . .
Patterns: *N-be-to*	**Patterns**: *N-be-that/of*
Thing **family**	***Reason*** **family**
Semantic features: [factual]	**Semantic features**: [factual], [causal]
Frame: general factual	**Frame**: causal; attentional focus on CAUSE
Nouns: ***fact***, ***phenomenon***, *point*, *case*, *thing*, *business*	**Nouns**: ***reason***, *cause*, *ground*, *thing*
Patterns: *N-that*, *N-be-that*	**Patterns**: *N-be-that/why*, *N-that/why*

Table 3: Example families from Schmid (2000). The nouns in boldface are used to evaluate this work.

lowing Schmid's cataphoric patterns, 80% of the instances follow the pattern *N-to*.[6]

4.2 Categorization of shell nouns

Schmid classifies shell nouns at three levels. At the most abstract level, he classifies shell nouns into six semantic classes: *factual*, *linguistic*, *mental*, *modal*, *eventive*, and *circumstantial*. Each semantic class indicates the type of experience the shell noun is intended to describe. For instance, the *mental* class describes ideas and cognitive states, whereas the *linguistic* class describes utterances, linguistic acts, and products thereof.

The next level of classification includes more-detailed semantic features. Each broad semantic class is sub-categorized into a number of *groups*. A group of an abstract class tries to capture the semantic features associated with the fine-grained differences between different usages of shell nouns in that class. For instance, groups associated with the *mental* class are: *conceptual*, *creditive*, *dubiative*, *volitional*, and *emotive*.

The third level of classification consists of *families*. A family groups together shell nouns with similar semantic features. Schmid provides 79 distinct families of 670 shell nouns. Each family is named after the primary noun in that family. Table 3 shows six families: *Idea*, *Plan*, *Trouble*, *Problem*, *Thing*, and *Reason*. A shell noun can be a member of multiple families. The nouns subsumed in a family share semantic features. For instance, all nouns in the *Idea* family are *mental* and *conceptual*. They are mental because ideas are only accessible through thoughts, and conceptual because they represent reflection or an application of a concept. Each family activates a *semantic frame*. The idea of these semantic frames is similar to that of frames in Frame semantics (Fillmore, 1985) and in semantics of grammar (Talmy, 2000). In particular, Schmid follows Talmy's conception of frames. A semantic frame describes conceptual structures, its elements, and their interrelationships. For instance, the *Reason* family invokes the causal frame, which has cause and effect as its elements with the attentional focus on the cause. According to Schmid, the nouns in a family also share a number of lexico-syntactic features. The *patterns* attribute in Table 3 shows prototypical lexico-syntactic patterns, which *attract* the members of the family. Schmid defines *attraction* as the degree to which a lexico-grammatical pattern attracts a certain noun. For instance, the patterns *N-to* and *N-that* attract the shell nouns in the *Plan* family, whereas the *N-that* pattern attracts the nouns in the *Thing* family. The pattern *N-of* is restricted to a smaller group of nouns such as *concept*, *problem*, and *issue*.[7,8]

[6]Table 2 does not include anaphoric patterns, as this paper is focused on cataphoric shell noun examples. Anaphoric patterns are common for all shell nouns: among all instances of a shell noun, approximately 50 to 80% are anaphoric.

[7]Schmid used the British section of COBUILD'S *Bank of English* for his classification.

[8]Schmid's families could help enrich resources such as FrameNet (Baker et al., 1998) with the shell content information.

5 Resolution algorithm

With this exposition, the problem of shell noun resolution is identifying the appropriate semantic argument of the shell noun representing its shell content. This section describes our algorithm to resolve shell nouns following cataphoric patterns. The algorithm addresses the primary challenge of idiosyncrasies of shell nouns by exploiting Schmid's semantic families (see Section 4.2). The input of the algorithm is a shell noun instance following a cataphoric pattern, and the output is its shell content or *None* if the shell content is not present in the given sentence. The algorithm follows three steps. First, we parse the given sentence using the Stanford parser.[9] Second, we look for the noun phrase (NP), where the head of the NP is the shell noun to be resolved.[10] Finally, we extract the appropriate shell content, if it is present in the given sentence.

5.1 Identifying potentially anaphoric shell-noun constructions

Before starting the actual resolution, first we identify whether the shell content occurs in the given sentence or not. According to Schmid, the lexico-syntactic patterns signal the position of the shell content. For instance, if the pattern is of the form *N-be-clause*, the shell content is more likely to occur in the complement clause in the same sentence. That said, although on the surface level, the shell noun seems to follow a cataphoric pattern, it is possible that the shell content is not given in a postnominal or a complement clause, as shown in (7).

> **(7)** Just as weekend hackers flock to the golf ball most used by PGA Tour players, **recreational skiers, and a legion of youth league racers, gravitate to the skis worn by Olympic champions**. It is <u>**the reason**</u> that top racers are so quick flash their skis for the cameras in the finish area.

Here, the shell noun and its content are linked via the pronoun *it*. For such constructions, the shell noun phrase and shell content do not occur in the same sentence. Shell content occurs in the preceding discourse, typically in the preceding sentence. We identify such cases, and other cases where the shell content is not likely to occur in the postnominal or complements clauses, by looking for the patterns below in the given order, returning the shell content when it occurs in the given sentence.

Sub-be-N This pattern corresponds to the lexico-grammatical pattern in Figure 1(a). If this pattern is found, there are three main possibilities for the *subject*. First, if an existential *there* occurs at the subject position, we move to the next pattern. Second, if the subject is *it* (example (7)), *this* or *that*, we return *None*, assuming that the content is not present in the given sentence. Finally, if the first two conditions are not satisfied, i.e., if the subject is neither a pronoun not an existential *there*, we assume that subject contains a valid shell content, and return it. An example is shown in (8). Note that in such cases, unlike other patterns, the shell content is expressed as a noun phrase.

> **(8)** **Strict liability** is <u>**the biggest issue**</u> when considering what athletes put in their bodies.

Apposition Another case where shell content does not typically occur in the postnominal or complement clause is the case of apposition. Indefinite shell noun phrases often occur in apposition constructions, as shown in (9).

> **(9)** The LH lineup, according to Gale, will feature **"cab-forward" design**, <u>**a concept**</u> that particularly pleases him.

In this step, we check for this construction and return the sentential, verbal, or nominal left sibling of the shell noun phrase.

Modifier For shell nouns such as *issue*, *phenomenon*, and *policy*, often the shell content is given in the modifier of the shell noun, as shown in (10).

> **(10)** But in the 18th century, Leipzig's central location in German-speaking Europe and **the liberal trade** <u>**policy**</u> of the Saxon court fostered publishing.

We deal with such cases as follows. First, we extract the modifier phrases by concatenating the modifier words having noun, verb, or adjective part-of-speech tags. To exclude unlikely modifier phrases as shell content (e.g., *good idea*, *big*

[9] `http://nlp.stanford.edu/software/lex-parser.shtml`

[10] We extract the head of an NP following the heuristics proposed by Collins (1999, p. 238).

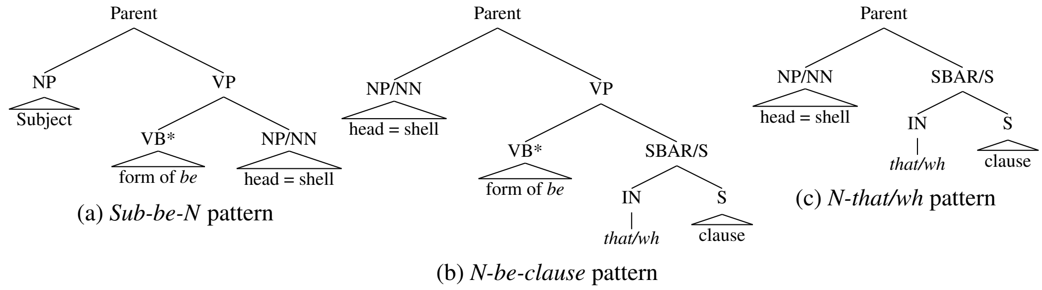

Figure 1: Lexico-syntactic patterns for shell nouns

issue), we extract a list of modifiers for a number of shell nouns and create a stoplist of modifiers. If any of the words in the modifier phrases is a pronoun or occurs in the stoplist, we move to the next pattern. If the modifier phrase passes the stoplist test, to distinguish between non-shell content and shell content modifiers, we examine the hypernym paths of the words in the modifier phrase in WordNet (Fellbaum, 1998). If the synset *abstraction.n.06* occurs in the path, we consider the modifier phrase to be valid shell content, assuming that the shell content of shell nouns most typically represents an abstract entity.

5.2 Resolving remaining instances

At this stage we are assuming that the shell content occurs either in the postnominal clause or the complement clause. So we look for the patterns below, returning the shell content when found.

N-be-clause The lexico-grammatical pattern corresponding to the pattern *N-be-clause* is shown in Figure 1(b). This is one of the more reliable patterns for shell content extraction, as the *be* verb suggests the semantic identity between the shell noun and the complement clause. The *be*-verb does not necessarily have to immediately follow the shell noun. For instance, in example (2), the head of the NP *The issue that this country and Congress must address* is the shell noun *issue*, and hence it satisfies the construction in Figure 1(b).

N-clause Finally, we look for this pattern. An example of this pattern is shown in Figure 1(c). This is the most common (see Table 2) and trickiest pattern in terms of resolution, and whether the shell content is given in the postnominal clause or not is dependent on the properties of the shell noun under consideration and the syntactic construction of the example. For instance, for the shell noun *decision*, the postnominal *to*-infinitive clause typically represents shell content. But this did not hold for the shell noun *reason*, as shown in (11).

(11) **The reason** to resist becoming a participant is obvious.

Here, Schmid's semantic families come in the picture. We wanted to examine a) the extent to which the previous steps help in resolution, and b) whether knowledge extracted from Schmid's families add value to the resolution. So we employ two versions of this step.

Include Schmid's cues (+SC) This version exploits the knowledge encoded in Schmid's semantic families (Section 4.2), and extracts postnominal clauses only if Schmid's pattern cues are satisfied. In particular, given a shell noun, we determine the families in which it occurs and list all possible patterns of these families as shell content cues. The postnominal clause is a valid shell content only if it satisfies these cues. For instance, the shell noun *reason* occurs in only one family: *Reason*, with the allowed shell content patterns *N-that* and *N-why*. Schmid's patterns suggest that the postnominal *to*-infinitive clauses are not allowed as shell content for this shell noun, and thus this step will return *None*. This version helps correctly resolving examples such as (11) to *None*.

Exclude Schmid's cues (–SC) This version does not enforce Schmid's cues in extracting the postnominal clauses. For instance, the *Problem* family does not include *N-that/wh/to/of* patterns, but in this condition, we nonetheless allow these patterns in extracting the shell content of the nouns from this family.

6 Evaluation data

We claim that our algorithm is able to resolve a variety of shell nouns. That said, creating evaluation data for all of Schmid's 670 English shell nouns is extremely time-consuming, and is therefore not pursued further in the current study. Instead we create a sample of representative evaluation data to examine how well the algorithm works a) on a variety of shell nouns, b) for shell nouns within a family, c) for shell nouns across families with completely different semantic and syntactic expectations, and d) for a variety of shell patterns from Table 1.

6.1 Selection of nouns

Recall that each shell noun has its idiosyncrasies. So in order to evaluate whether our algorithm is able to address these idiosyncrasies, the evaluation data must contain a variety of shell nouns with different semantic and syntactic expectations. To examine a), we consider the six families shown in Table 3. These families span three abstract categories: *mental*, *eventive*, and *factual*, and five distinct groups: *conceptual*, *volitional*, *factual*, *causal*, and *attitudinal*. Also, the families have considerably different syntactic expectations. For instance, the nouns in the *Idea* family can have their content in *that* or *of* clauses occurring in *N-clause* or *N-be-clause* constructions, whereas the *Trouble* and *Problem* families do not allow *N-clause* pattern. The shell content of the nouns in the *Plan* family is generally represented with *to*-infinitive clauses. To examine b) and c), we choose three nouns from each of the first four families from Table 3. To add diversity, we also include two shell nouns from the *Thing* family and a shell noun from the *Reason* family. So we selected a total of 12 shell nouns for evaluation: *idea*, *issue*, *concept*, *decision*, *plan*, *policy*, *problem*, *trouble*, *difficulty*, *reason*, *fact*, and *phenomenon*.

6.2 Selection of instances

Recall that the shell content varies based on the shell noun and the pattern it follows. Moreover, shell nouns have pattern preferences, as shown in Table 2. To examine d), we need shell noun examples following different patterns from Table 1. We consider the New York Times corpus as our base corpus, and from this corpus extract all sentences following the lexico-grammatical patterns in Table 1 for the twelve selected shell nouns. Then we arbitrarily pick 100 examples for each shell noun, making sure that the selection contains examples of each cataphoric pattern from Table 1. These examples consist of 70% examples of each of the seven cataphoric patterns, and the remaining 30% of the examples are picked randomly from the distribution of patterns for that shell noun.

6.3 Crowdsourcing annotation

We designed a crowdsourcing experiment to obtain the annotated data for evaluation. We parse each sentence using the Stanford parser, and extract all possible candidates, i.e., arguments of the shell noun from the parser's output. Since our examples include embedding clauses and sentential complements, the parser is often inaccurate. For instance, in example (12), the parser attaches only the first clause of the coordination (*that people were misled*) to the shell noun *fact*.

(12) **<u>The fact</u> that people were misled and information was denied**, that's the reason that you'd wind up suing.

To deal with such parsing errors, we consider the 30-best parses given by the parser. From these parses, we extract a list of eligible candidates. This list includes the arguments of the shell noun given in the appositional clauses, modifier phrases, postnominal *that*, *wh*, or *to*-infinitive clauses, complement clauses, objects of postnominal prepositions of the shell noun, and subject if the shell noun follows *subject-be-N* construction. On average, there were three candidates per instance.

After extracting the candidates, we present the annotators with the sentence, with the shell noun highlighted, and the extracted candidates. We ask the annotators to choose the option that provides the correct interpretation of the highlighted shell noun. We also provide them the option *None of the above*, and ask them to select it if the shell content is not present in the given sentence or the shell content is not listed in the list of candidates.

CrowdFlower We used CrowdFlower[11] as our crowdsourcing platform, which in turn uses various worker channels such as Amazon Mechanical Turk[12]. CrowdFlower offers a number of features. First, it provides a *quiz* mode which facilitates filtering out spammers by requiring an annotator to pass a certain number of test questions before starting the real annotation. Second, during annotation, it randomly presents test questions with known answers to the annotators to keep them on their toes. Based on annotators' responses to the test questions, each annotator is assigned a trust

[11] http://crowdflower.com/

[12] https://www.mturk.com/mturk/welcome

	≥ 5	≥ 4	≥ 3	< 3
idea	53	67	95	5
issue	44	65	95	5
concept	40	56	96	4
decision	50	72	98	2
plan	41	55	95	5
policy	42	61	94	6
problem	52	70	100	0
trouble	44	69	99	1
difficulty	45	61	96	4
reason	48	60	93	7
fact	52	68	98	2
phenomenon	39	56	95	5
all	46	63	96	4

Table 4: Annotator agreement on shell content. Each column shows the percentage of instances on which at least n or fewer than n annotators agree on a single answer.

	Nouns	LSC	A–SC	A+SC
1	*idea*	74	82	**83**
2	*issue*	60	75	**77**
3	*concept*	51	67	**68**
4	*decision*	70	71	**73**
5	*plan*	51	**63**	62
6	*policy*	58	**70**	52
7	*problem*	66	**69**	59
8	*trouble*	63	**68**	50
9	*difficulty*	68	**75**	49
10	*reason*	43	53	**77**
11	*fact*	43	55	**68**
12	*phenomenon*	33	**62**	50
13	*all*	57	**69**	64

Table 5: Shell noun resolution results. Each column shows the percent accuracy of resolution with the respective method. Boldface is best in row.

score: an annotator performing well on the test questions gets a high trust score. Finally, CrowdFlower allows the user to select the permitted demographic areas and skills required.

Settings We asked for at least 5 annotations per instance by annotators from the English-speaking countries. The evaluation task contained a total of 1200 instances, 100 instances per shell noun. To maintain the annotation quality, we included 105 test questions, distributed among different answers. We paid 2.5 cents per instance and the annotation task was completed in less than 24 hours.

Results Table 4 shows the agreement of the crowd. In most cases, at least 3 out of 5 annotators agreed on a single answer. We took this answer as the gold standard in our evaluation, and discard the instances where fewer than three annotators agreed. The option *None of the above* was annotated for about 30% of the cases. We include these cases in the evaluation. In total we had 1,257 instances (1,152 instances where at least 3 annotators agreed + 105 test questions).

7 Evaluation results

Baseline We evaluate our algorithm against crowd-annotated data using a *lexico-syntactic clause* (LSC) baseline. Given a sentence containing a shell instance and its parse tree, this baseline extracts the postnominal or complement clause from the parse tree depending only upon the lexico-syntactic pattern of the shell noun. For instance, for the *N-that* and *N-be-to* patterns, it extracts the postnominal *that* clause and the complement *to*-infinitive clause, respectively.[13]

Results Table 5 shows the evaluation results for the LSC baseline, the algorithm without Schmid's cues (A–SC), and the algorithm with Schmid's cues (A+SC). The A–SC condition in all cases and the A+SC condition in some cases outperform the LSC baseline, which proves to be rather low, especially for the shell nouns with strict syntactic expectations (e.g., *fact* and *reason*). Thus we see that our algorithm is adding value.

That said, we observe a wide range of performance for different shell nouns. On the up side, adding Schmid's cues helps resolving the shell nouns with strict syntactic expectations. The A+SC results for the shell nouns *idea*, *issue*, *concept*, *decision*, *reason*, and *fact* outperform the baseline and the A–SC results. In particular, the A+SC results for the shell nouns *fact* and *reason* are markedly better than the baseline results. These nouns have strict syntactic expectations for the shell content clauses they take: the families *Thing* and *Certainty* of the shell noun *fact* allow only a *that* clause, and the *Reason* family of the shell noun *reason* allows only *that* and *because* clauses for the shell content. These cues help in correctly resolving examples such as (11) to *None*, where the postnominal *to*-infinitive clause

[13]Note that we only extract subordinating clauses (e.g., (SBAR (IN *that*) (*clause*))) and *to*-infinitive clauses, and not relative clauses.

describes the purpose or the goal for the reason, but not the shell content itself.

On the down side, adding Schmid's cues hurts the performance of more versatile nouns, which can take a variety of clauses. Although the A–SC results for the shell nouns *plan*, *policy*, *problem*, *trouble*, *difficulty*, and *phenomenon* are well above the baseline, the A+SC results are markedly below it. That is, Schmid's cues were deleterious. Our error analysis revealed that these nouns are versatile in terms of the clauses they take as shell content, and Schmid's cues restrict these clauses to be selected as shell content. For instance, the shell noun *problem* occurs in two semantic families with *N-be-that/of* and *N-be-to* as pattern cues (Table 3), and postnominal clauses are not allowed for this noun. Although these cues help in filtering some unwanted cases, we observed a large number of cases where the shell content is given in postnominal clauses, as shown in (13).

(13) I was trying to address **<u>the problem</u>** of **unreliable testimony by experts** in capital cases.

Similarly, the *Plan* family does not allow the *N-of* pattern. This cue works well for the shell noun *decision* from the same family because often the postnominal *of* clause is the agent for this shell noun and not the shell content. However, it hurts the performance of the shell noun *policy*, as *N-of* is a common pattern for this shell noun (e.g., *... officials in Rwanda have established* ***<u>a policy</u>*** *of* ***refusing to protect refugees...***). Other failures of the algorithm are due to parsing errors and lack of inclusion of context information.

8 Discussion and conclusion

In this paper, we proposed a general method to resolve shell nouns following cataphoric constructions. This is a first step towards end-to-end shell noun resolution. In particular, this method can be used to create training data for any given shell noun, which can later be used to resolve harder anaphoric cases of that noun using the method that we proposed earlier (Kolhatkar et al., 2013b).

The first goal of this work was to point out the difficulties associated with the resolution of cataphoric cases of shell nouns. The low resolution results of the LSC baseline demonstrate the difficulties of resolving such cases using syntax alone, suggesting the need for incorporating more linguistic knowledge in the resolution.

The second goal of this work was to examine to what extent knowledge derived from the linguistics literature helps in resolving shell nouns. We conclude that Schmid's pattern and clausal cues are useful for resolving nouns with strict syntactic expectations (e.g., *fact*, *reason*); however, these cues are defeasible: they miss a number of cases in our corpus. It is possible to improve on Schmid's cues using crowdsourcing annotation and by exploiting lexico-syntactic patterns associated with different shell nouns from a variety of corpora.

One limitation of our approach is that in our resolution framework, we do not consider the problem of ambiguity of nouns that might not be used as shell nouns. The occurrence of nouns with the lexical patterns in Table 1 does not always guarantee shell noun usage. For instance, in our data, we observed a number of instances of the noun *issue* with the publication sense (e.g., *this week's issue of Sports Illustrated*).

Our algorithm is able to deal with only a restricted number of shell noun usage constructions, but the shell content can be expressed in a variety of other constructions. A robust machine learning approach that incorporates context and deeper semantics of the sentence, along with Schmid's cues, could mitigate this limitation.

This work opens a number of new research directions. Our next planned task is clustering different shell nouns based on the kind of complements they take in different usages similar to verb clustering (Merlo and Stevenson, 2000; Schulte im Walde and Brew, 2002).

Acknowledgements

We thank the anonymous reviewers for their comments. We also thank Suzanne Stevenson, Gerald Penn, Heike Zinsmeister, Kathleen Fraser, Aida Nematzadeh, and Ryan Kiros for their feedback. This research was financially supported by the Natural Sciences and Engineering Research Council of Canada and by the University of Toronto.

References

Nicholas Asher. 1993. *Reference to Abstract Objects in Discourse*. Kluwer Academic Publishers, Dordrecht, Netherlands.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet Project. In *Proceed-*

ings of the 17th International Conference on Computational Linguistics, volume 1 of *COLING '98*, pages 86–90, Stroudsburg, PA, USA. Association for Computational Linguistics.

Douglas Biber, Stig Johansson, Geoffrey Leech, Susan Conrad, and Edward Finegan. 1999. *Longman Grammar of Spoken and Written English*. Pearson ESL, November.

Simon Philip Botley. 2006. Indirect anaphora: Testing the limits of corpus-based linguistics. *International Journal of Corpus Linguistics*, 11(1):73–112.

Donna K. Byron. 2003. Annotation of pronouns and their antecedents: A comparison of two domains. Technical report, University of Rochester. Computer Science Department.

Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania.

Miriam Eckert and Michael Strube. 2000. Dialogue acts, synchronizing units, and anaphora resolution. *Journal of Semantics*, 17:51–89.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

Charles J. Fillmore. 1985. Frames and the semantics of understanding. *Quaderni di Semantica*, 6(2):222–254.

John Flowerdew. 2003. Signalling nouns in discourse. *English for Specific Purposes*, 22(4):329–346.

Gill Francis. 1994. Labelling discourse: An aspect of nominal group lexical cohesion. In M. Coulthard, editor, *Advances in written text analysis*, pages 83–101. Routledge, London.

Matthew Gerber, Joyce Chai, and Adam Meyers. 2009. The role of implicit argumentation in nominal srl. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 146–154, Boulder, Colorado, June. Association for Computational Linguistics.

Nancy Hedberg, Jeanette K. Gundel, and Ron Zacharski. 2007. Directly and indirectly anaphoric demonstrative and personal pronouns in newspaper articles. In *Proceedings of DAARC-2007 8th Discourse Anaphora and Anaphora Resolution Colloquium*, pages 31–36.

Eli Hinkel. 2004. *Teaching Academic ESL Writing: Practical Techniques in Vocabulary and Grammar (ESL and Applied Linguistics Professional)*. Lawrence Erlbaum, Mahwah, NJ, London.

Rodney D. Huddleston and Geoffrey K. Pullum. 2002. *The Cambridge Grammar of the English Language*. Cambridge University Press, April.

Roz Ivanic. 1991. Nouns in search of a context: A study of nouns with both open- and closed-system characteristics. *International Review of Applied Linguistics in Language Teaching*, 29:93–114.

Varada Kolhatkar, Heike Zinsmeister, and Graeme Hirst. 2013a. Annotating anaphoric shell nouns with their antecedents. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 112–121, Sofia, Bulgaria, August. Association for Computational Linguistics.

Varada Kolhatkar, Heike Zinsmeister, and Graeme Hirst. 2013b. Interpreting anaphoric shell nouns using antecedents of cataphoric shell nouns as training data. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 300–310, Seattle, Washington, USA, October. Association for Computational Linguistics.

Paola Merlo and Suzanne Stevenson. 2000. Automatic verb classification based on statistical distributions of argument structure. *Computational Linguistics*, 27(3):373–408.

Adam Meyers, Ruth Reeves, Catherine Macleod, Rachel Szekely, Veronika Zielinska, Brian Young, and Ralph Grishman. 2004. The nombank project: An interim report. In *In Proceedings of the NAACL/HLT Workshop on Frontiers in Corpus Annotation*.

Christoph Müller. 2008. *Fully Automatic Resolution of It, This and That in Unrestricted Multi-Party Dialog*. Ph.D. thesis, Universität Tübingen.

Costanza Navarretta. 2011. Antecedent and referent types of abstract pronominal anaphora. In *Proceedings of the Workshop Beyond Semantics: Corpus-based investigations of pragmatic and discourse phenomena*, Göttingen, Germany, Feb.

Rebecca J. Passonneau. 1989. Getting at discourse referents. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, pages 51–59, Vancouver, British Columbia, Canada. Association for Computational Linguistics.

Massimo Poesio and Ron Artstein. 2008. Anaphoric annotation in the ARRAU corpus. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, May. European Language Resources Association (ELRA).

Massimo Poesio, Simone Ponzetto, and Yannick Versley. 2011. Computational models of anaphora resolution: A survey. Unpublished.

Judita Preiss, Ted Briscoe, and Anna Korhonen. 2007. A system for large-scale acquisition of verbal, nominal and adjectival subcategorization frames from corpora. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 912–919, Prague, Czech Republic, June. Association for Computational Linguistics.

Hans-Jörg Schmid. 2000. *English Abstract Nouns As Conceptual Shells: From Corpus to Cognition*. Topics in English Linguistics 34. Mouton de Gruyter, Berlin.

Sabine Schulte im Walde and Chris Brew. 2002. Inducing German semantic verb classes from purely syntactic subcategorisation information. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 223–230, Philadelphia, PA.

Leonard Talmy. 2000. The windowing of attention. In *Toward a Cognitive Semantics*, volume 1, pages 257–309. The MIT Press.

Zeno Vendler. 1968. *Adjectives and Nominalizations.* Mouton and Co., The Netherlands.